University of Brasília

Institute of Exact Sciences
Department of Computer Science

Deep Learning & Remote Sensing: Pushing the Frontiers in Image Segmentation

Osmar Luiz Ferreira de Carvalho

Thesis presented in partial fulfillment of the requirements for the degree of Master of Science in Informatics

Advisor
Prof. Dr. Díbio Leandro Borges

Brazil
2022

University of Brasília

Institute of Exact Sciences
Department of Computer Science

Deep Learning & Remote Sensing: Pushing the Frontiers in Image Segmentation

Osmar Luiz Ferreira de Carvalho

Thesis presented in partial fulfillment of the requirements for the degree
of Master of Science in Informatics

Prof. Dr. Díbio Leandro Borges (Advisor)
CIC/UnB

Prof. Dr. Eraldo Aparecido Trondoli Matricardi
University of Brasília, Brazil

Prof. Dr. Yosio Edemir Shimabukuro
Instituto Nacional de Pesquisas Espaciais, Brazil

Prof. Dr. Ricardo Pezzuol Jacobi
Coordinator of the Graduate Program in Informatics

Brazil, Brasília, April 5 2022

Dedication

To the reader.

Agradecimentos

Ao meu núcleo familiar, que nesse periodo de pandemia, foram cruciais e acompanharam o desenvolvimento dos artigos e todas as instâncias da minha formação, me passando valores inestimaveis, um deles o de valorização do conhecimento. Nada disso teria sido possível sem eles. Ainda ressalto que tenho realizado um sonho trabalhar com o meu pai, que foi um grande apoio em todos os trabalhos desenvolvidos.

Ao restante da minha família, que fizeram com que esse periodo tivesse sido mais fácil e tranquilo. Seria exaustivo enunciar a todos, mas expresso minha mais profunda gratidão.

À minha namorada Natalia, pelo amor, apoio e suporte incondicional, especialmente nesse período difícil da pandemia, que me passou tranquilidade e animo para fazer essa dissertação.

Ao meu orientador Dibio Leandro Borges, que sempre foi muito parceiro, solícito e ajudou imensamente no desenvolvimento dos artigos.

Aos membros da Yosio Shimabukuro e Eraldo Matricardi, que sempre foram muito solícitos e forneceram valiosas contribuições.

Aos meus muitos amigos que sempre me incentivaram e me deram suporte ao longo de toda a minha vida. Em especial aos meus amigos do meu semestre que fundaram o capítulo estudantil IEEE CIS comigo em 2018 (Pedro, Thiago, Lins, Seiki, Allan, Lins, Marcelo). As atividades que começamos a realizar naquele periodo foram cruciais para que eu achasse um campo aonde eu me sentisse tão feliz e realizado.

Ao LSIE pela infraestrutura e em especial ao meu querido amigo Anesmar pela amizade e companheirimo em todas as pesquisas feitas, sendo uma peça fundamental da nossa equipe.

À ANEEL, em especial ao Issao e o Alex, que forneceram a minha primeira oportunidade de estágio ainda durante a minha graduação e que os trabalhos perduram até hoje, rendendo um dos artigos desta dissertação.

À SPU, que forneceram financiamento e dados cruciais para a elaboração de dois dos trabalhos desta dissertação.

À SEDUH, que se dispuseram a disponibilizar online os resultados obtidos em um dos trabalhos desenvolvidos.

Ao PPGI, que forneceu disciplinas e professores que de alguma forma ajudaram no desenvolvimento da dissertação.

"We cannot solve our problems with the same thinking we used when we created them."

Albert Einstein

Abstract

Image segmentation aims to simplify the understanding of digital images. Deep learning-based methods using convolutional neural networks have been game-changing, allowing the exploration of different tasks (e.g., semantic, instance, and panoptic segmentation). Semantic segmentation assigns a class to every pixel in an image, instance segmentation classifies objects at a pixel level with a unique identifier for each target, and panoptic segmentation combines instance-level predictions with different backgrounds. Remote sensing data largely benefits from those methods, being very suitable for developing new DL algorithms and creating solutions using top-view images. However, some peculiarities prevent remote sensing using orbital and aerial imagery from growing when compared to traditional ground-level images (e.g., camera photos): (1) The images are extensive, (2) it presents different characteristics (e.g., number of channels and image format), (3) a high number of pre-processes and post-processes steps (e.g., extracting patches and classifying large scenes), and (4) most open software for labeling and deep learning applications are not friendly to remote sensing due to the aforementioned reasons. This dissertation aimed to improve all three main categories of image segmentation. Within the instance segmentation domain, we proposed three experiments. First, we enhanced the box-based instance segmentation approach for classifying large scenes, allowing practical pipelines to be implemented. Second, we created a bounding-box free method to reach instance segmentation results by using semantic segmentation models in a scenario with sparse objects. Third, we improved the previous method for crowded scenes and developed the first study considering semi-supervised learning using remote sensing and GIS data. Subsequently, in the panoptic segmentation domain, we presented the first remote sensing panoptic segmentation dataset containing fourteen classes and disposed of software and methodology for converting GIS data into the panoptic segmentation format. Since our first study considered RGB images, we extended our approach to multispectral data. Finally, we leveraged the box-free method initially designed for instance segmentation to the panoptic segmentation task. This dissertation analyzed various segmentation methods and image types, and the developed solutions enable the exploration of new tasks (such as panoptic segmentation), the simplification of labeling data (using the proposed semi-supervised learning procedure), and a simplified way to obtain instance and panoptic predictions using simple semantic segmentation models.

Keywords: semantic segmentation, instance segmentation, panoptic segmentation, GIS, remote sensing, deep learning

Resumo

A segmentação de imagens visa simplificar o entendimento de imagens digitais e métodos de aprendizado profundo usando redes neurais convolucionais permitem a exploração de diferentes tarefas (e.g., segmentação semântica, instância e panóptica). A segmentação semântica atribui uma classe a cada pixel em uma imagem, a segmentação de instância classifica objetos a nível de pixel com um identificador exclusivo para cada alvo e a segmentação panóptica combina instâncias com diferentes planos de fundo. Os dados de sensoriamento remoto são muito adequados para desenvolver novos algoritmos. No entanto, algumas particularidades impedem que o sensoriamento remoto com imagens orbitais e aéreas cresça quando comparado às imagens tradicionais (e.g., fotos de celulares): (1) as imagens são muito extensas, (2) apresenta características diferentes (e.g., número de canais e formato de imagem), (3) um grande número de etapas de pré-processamento e pós-processamento (e.g., extração de quadros e classificação de cenas grandes) e (4) os softwares para rotulagem e treinamento de modelos não são compatíveis. Esta dissertação visa avançar nas três principais categorias de segmentação de imagens. Dentro do domínio de segmentação de instâncias, propusemos três experimentos. Primeiro, aprimoramos a abordagem de segmentação de instância baseada em caixa para classificar cenas grandes. Em segundo lugar, criamos um método sem caixas delimitadoras para alcançar resultados de segmentação de instâncias usando modelos de segmentação semântica em um cenário com objetos esparsos. Terceiro, aprimoramos o método anterior para cenas aglomeradas e desenvolvemos o primeiro estudo considerando aprendizado semissupervisionado usando sensoriamento remoto e dados GIS. Em seguida, no domínio da segmentação panóptica, apresentamos o primeiro conjunto de dados de segmentação panóptica de sensoriamento remoto e dispomos de uma metodologia para conversão de dados GIS no formato COCO. Como nosso primeiro estudo considerou imagens RGB, estendemos essa abordagem para dados multiespectrais. Por fim, melhoramos o método box-free inicialmente projetado para segmentação de instâncias para a tarefa de segmentação panóptica. Esta dissertação analisou vários métodos de segmentação e tipos de imagens, e as soluções desenvolvidas permitem a exploração de novas tarefas , a simplificação da rotulagem de dados e uma forma simplificada de obter previsões de instância e panópticas usando modelos simples de segmentação semântica.

Palavras-chave: segmentação semântica, segmentação de instâncias, segmentação panótica, GIS, sensoriamento remoto, aprendizagem profunda

Contents

List of Figures

List of Tables

List of Abbreviations and Acronyms

AI Artificial Intelligence

ML Machine Learning

DL Deep Learning

CNN Convolutional Neural Network

GIS Geographic Information Systems

COCO Common Objects in Context

JSON JavaScript Object Notation

RGB Red, Green, and Blue

RCNN Region Convolutional Neural Network

AP Average Precision

PQ Panoptic Quality

RQ Recognition Quality

SQ Segmentation Quality

AUC Area Under the Curve

ESA European Space Agency

CSV Comma Separated Value

GPU Graphics Processing Unit

Chapter 1

Introduction

We live in the Big Data era. Endless sources (e.g., satellites, smartphones, websites) generate raw data non-stop, rapidly feeding ever-larger databases. The information flow is so high that developing automated mechanisms is crucial to handle massive amounts of data. In this regard, high-quality structured data is vital to understanding patterns, guiding decision-making processes, and reducing laborious work.

Nowadays, the talk of the town when referring to automated processes is deep learning (DL). DL is a subsection of Artificial Intelligence (AI) that uses neural networks (Schmidhuber, 2015), a structure composed of weighted connections between neurons that iteratively learn high and low-level features such as textures and shapes through gradient descent (Nogueira et al., 2017). DL promoted a revolution in several fields of science, including visual recognition (Voulodimos et al., 2018), natural language processing (Sun et al., 2017; Young et al., 2018), speech recognition (Nassif et al., 2019; Zhang et al., 2018c) object detection (Sharma and Mir, 2020; Zhao et al., 2019) medical image analysis (Liu et al., 2019b; Serte et al., 2020; Zhou et al., 2019), person identification (Bharathi and Shamily, 2020; Kaur et al., 2020), among others.

Image recognition is one of the hottest topics regarding automation processes, being a fast-growing field. This success is mainly related to substantial growth in image sources (e.g., cameras, satellites, sensors), which enables the constitution of large databases and advances in convolutional neural networks (CNNs). The CNNs are a particular type of neural network that mimics the human frontal cortex - which allows understanding of the interaction of shapes, contours, and textures by applying convolutional kernels throughout the image resulting in feature maps, enabling low, medium, and high-level feature recognition (e.g., corners, parts of an object, and complete objects, respectively) (Nogueira et al., 2017). The usability in image processing is very high due to the CNN's ability to process data in multi-dimensional arrays (Lecun et al., 2015). Besides, the CNN's may present different configurations that enable the exploration of different tasks, from image classification to keypoint detection (Dhillon and Verma, 2020).

Remote sensing using aerial and orbital images is one of the most consistent data sources, providing periodic information with standardized characteristics (e.g., spatial resolution, number of bands). Those characteristics make it suitable for advancing and exploring computer vision deep learning models, and the increasing availability of satellite images alongside computational

improvements makes the remote sensing field conducive to using deep learning (DL) techniques (Ma et al., 2019).

Remote sensing top-view images presents peculiarities compared to ground-level images (e.g., cellphone pictures) that require specific solutions. First, the properties of the images are usually different, i.e., the images are much larger, there is a varying number of channels, and the image format is different. Most new deep learning methods are tested on well-known large datasets such as COCO (Lin et al., 2014), Mapillary Vistas (Neuhold et al., 2017), and Cityscapes (Cordts et al., 2016). All of those datasets use RGB images. Besides, the size of remote sensing images also requires post-processing steps using sliding windows.

Those previously mentioned time and consuming labor tasks have prevented the growth and exploration of various segmentation problems in the remote sensing field. First, most available software, such as the Facebook‘s Detectron2 (Wu et al., 2019) are designed for RGB images. The development of novel methods that uses more or less spectral bands is little explored in the instance, and panoptic segmentation, in which the first study using instance segmentation with more than three bands was only done last year (de Carvalho et al., 2021c). Panoptic segmentation has great potential in the remote sensing field, but there are still no works on this topic. Finally, most annotation tools are not designed for remote sensing images, difficulting the generation and exploration of new datasets.

1.1 Objectives

The present study aims to explore the three main segmentation tasks using remote sensing images, with the following secondary objectives:

- leverage the mechanism of the sliding window to box-based instance segmentation methods.
- investigate the impact of image dimensions on small object studies.
- design a semi-iterative learning procedure associated with GIS software to easily augment datasets.
- propose a box-free instance segmentation approach for touching and non-touching targets at a pixel level.
- propose the first panoptic segmentation dataset in the remote sensing field.
- propose a software that converts GIS shapefile data into the panoptic segmentation COCO annotation format.
- propose an analysis on multispectral data and panoptic segmentation.
- A simplification of the panoptic segmentation task as an extention of the semantic segmentation task.

1.2 Overview of this Dissertation

This manuscript was divided into eight chapters to address the previously stated research objectives. Chapter 2 presents background information to the readers regarding the most critical topics in this paper. The readers will find deep learning, CNN, and image segmentation information. Next, chapter 3 starts with the most traditional instance segmentation method, the Mask-RCNN. We propose an analysis of multispectral data, small objects, and an enhancement of the process for classifying large areas. Chapter 4 propose a box-free method for obtaining instance segmentation results for sparse (non-touching) objects, and Chapter 5 generalizes this method for crowded objects and explores the concepts of semi-supervised learning. Chapter 6 proposes the first panoptic segmentation approach in the remote sensing field. We proposed the first panoptic segmentation dataset and software to convert GIS data into the panoptic segmentation COCO format. Still, in the panoptic landscape, we proposed in chapter 7 the first approach of panoptic segmentation using multispectral data, and Chapter 8 presents an alternative approach using a box-free-based panoptic segmentation approach, being an extension of the proposed method in chapter 4 and 5. Finally, Chapter 9 brings the conclusions of this dissertation.

Chapter 2

Background

This chapter aims to connect the readers with the most critical topics that will be developed and discussed in the following chapters. In this regard, we explain deep learning and its foundations, the functioning of CNNs, the application of CNNs for image segmentation and its variants, the metrics used for those tasks, and the more commonly used remote sensing sensors with their specifications.

2.1 Deep Learning

2.1.1 Overview

The terms artificial intelligence, machine learning, and deep learning often cause confusion, which is understandable since all of them are related (Bengio et al., 2017; Bini, 2018; Goodfellow et al., 2016). In a nutshell, deep learning is a subgroup of machine learning, and machine learning is a subgroup of artificial intelligence (Figure 2.1). In other words, there is a hierarchical structure among them, and deep learning is the most specific.

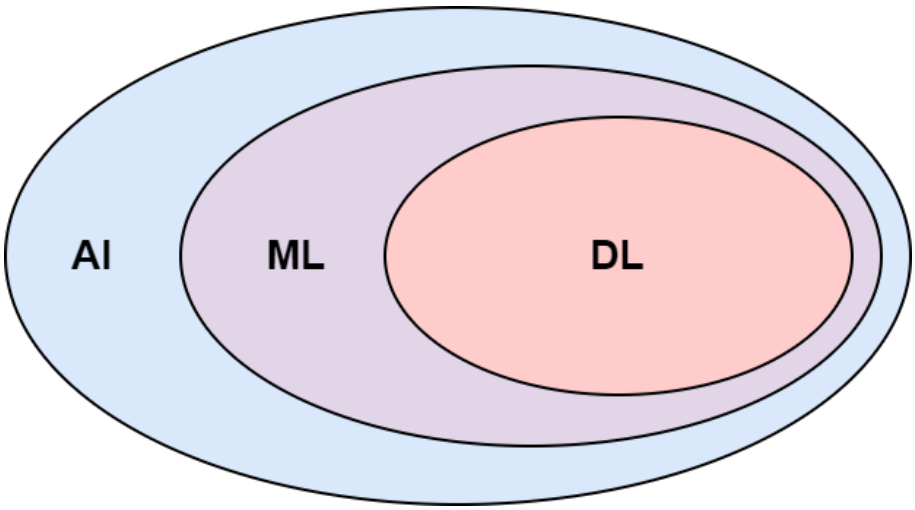

Figure (2.1) Hierarchical representation of Artificial Intelligence, Machine Learning, and Deep Learning.

Even though there is massive hype about AI, primarily due to the success of many deep learning applications, it has been around for a long time. AI is simply defined as a machine's ability to mimic human behavior. The thing is, the first implementations used manual configurations. In other words, the programs had a set of instructions made by humans, and all intelligence in the systems is predefined (Bengio et al., 2017).

Eventually, scientists realized that machines could figure out some information themselves. This is how Machine Learning (ML) started (El Naqa and Murphy, 2015). In this regard, a hand-designed set of rules now became a hand-designed selection of features of which a program with statistical models will learn and then define an output. The reader should understand features as pieces of important information.

Furthermore, deep learning emerges as a more complex solution (Lecun et al., 2015). The program learns the features that are relevant to the problem at hand. This happens because of the interaction within neurons present in different layers. This allows the systems to learn from other features at different levels of abstraction. Usually, the number of neurons and connections is so high that the entire pipeline is often seen as a black box.

2.1.2 Neural Networks

Neural networks encompass a large set of topics, having different variants. For example, CNN's are primarily used for image recognition, and Transformers and Long-short term memory have suitable applications with text. One of the simplest forms is the Multilayer Perceptron (MLP) (Bengio et al., 2017; Murtagh, 1991). The MLP structure has three-layer categories: input layer, hidden layers, and output layer (Figure 2.2). Each layer is composed of neurons connected to all neurons in the subsequent layer. The number of hidden layers and neurons may vary according to the programmer's specifications. Each neuron holds a value. The neurons in the first layer are often the raw data's value. Also, each neuron has connections with all neurons in the next layer. Mathematically, we can express each neuron as a simple function:

$$\sum_{i=i}^{m} w_i x_i + K, \tag{2.1}$$

in which m is the total number of neurons connected to this neuron, i is the instance of each neuron, w is the weight value, x is the input value of the previous neuron, and K is the bias.

However, the mathematical formulation can be improved. Since the number of connections is often very high, summing many weights and biases may yield extremely large results, propagating to the next layer, and diverging the entire system. Besides, up until now, all of the formulations are linear, which could be reduced and simplified. To solve this issue, we use activation functions. The activation function is basically a way to squish the values into a determined range of values, bringing non-linearity to the entire system and enabling the understanding of more complex patterns. There are many ways to establish this range of values. Table 2.1 lists commonly used activation functions.

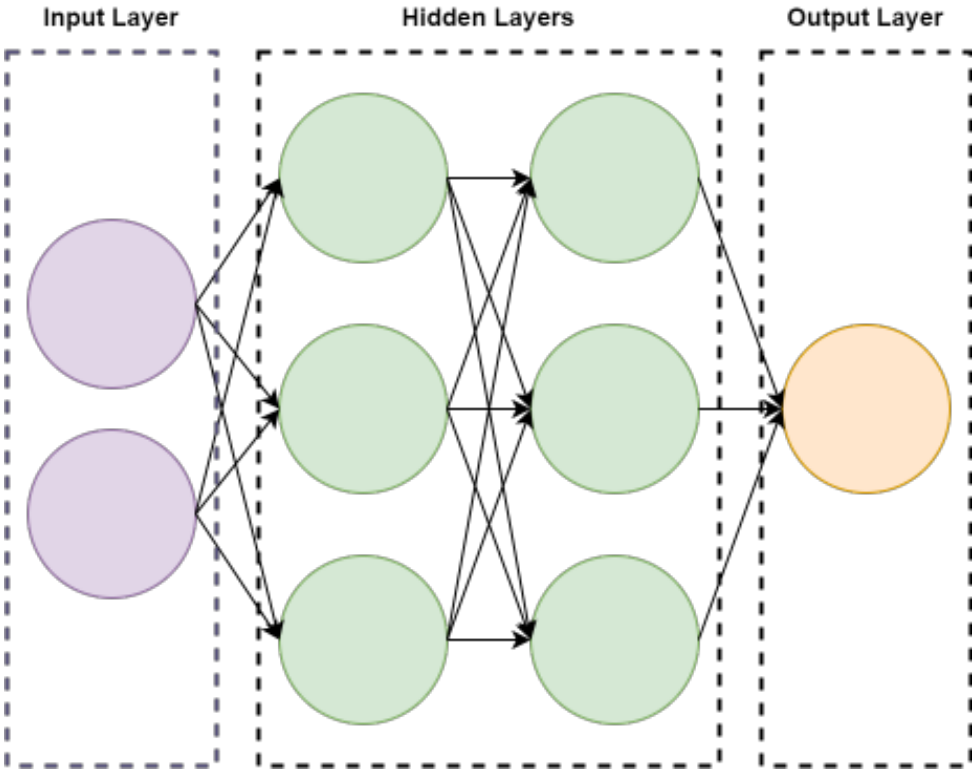

Figure (2.2) Simplified representation of the Multilayer Perceptron.

Table (2.1) Data split in the training validation and testing sets with their respective number of images and instances.

Name	Function	Range
binary	$\begin{cases} 0 \ if \ x \leq 0 \\ 1 \ if \ x > 0 \end{cases}$	0,1
sigmoid	$\sigma(x) = \frac{1}{1+e^{-x}}$	0,1
tanh	$tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$	-1,1
ReLU	$\begin{cases} 0 \ if \ x \leq 0 \\ x \ if \ x > 0 \end{cases}$	$0, \infty$
Leaky ReLU	$\begin{cases} 0.01x \ if \ x \leq 0 \\ x \ if \ x > 0 \end{cases}$	$-\infty, \infty$
Gaussian	e^{x^2}	0,1

So, basically, we can change the weights and biases' values to bring the best results. Nonetheless, it is important to note that the way computers can deal with such an extensive amount of data quickly is by using matrices. In a hypothetical scenario, the weights and biases for a neuron would be computed as follows:

$$\sigma\left(\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{k,0} & w_{k,1} & \cdots & w_{k,n} \end{bmatrix} \begin{bmatrix} a_0^{(0)} \\ a_1^{(0)} \\ \vdots \\ a_n^{(0)} \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{bmatrix}\right)$$

To reach the best weights and biases, we must first decide what the neural network needs to learn. In other words, we need a label for each sample in a given dataset. For example, suppose the objective is to predict which pixels in an image represents cars. In that case, it is necessary to build a dataset in which we know the output of all samples to make predictions on unseen data in the future. Since neural networks deal with numbers, a conventional way is to assign positive labels to the presence of a condition and a negative label to the absence of the given condition. Thus, the pixels that belong to cars will receive a 1-label and non-cars a 0-label. If the expected result is known, it is straightforward to compute how well the neural network is adjusted to this data set. This can be done by calculating the difference between the predicted value and what the value was actually supposed to be, and if we could adjust the weights and biases to reduce the sum of all errors to a minimal value, this would be ideal. In this regard, this function is called the cost function of a neural network, which is simply a measure of the predicted and expected results. Many possible functions are to be used, and many studies propose new functions to better adjust to a specific dataset (Janocha and Czarnecki, 2017).

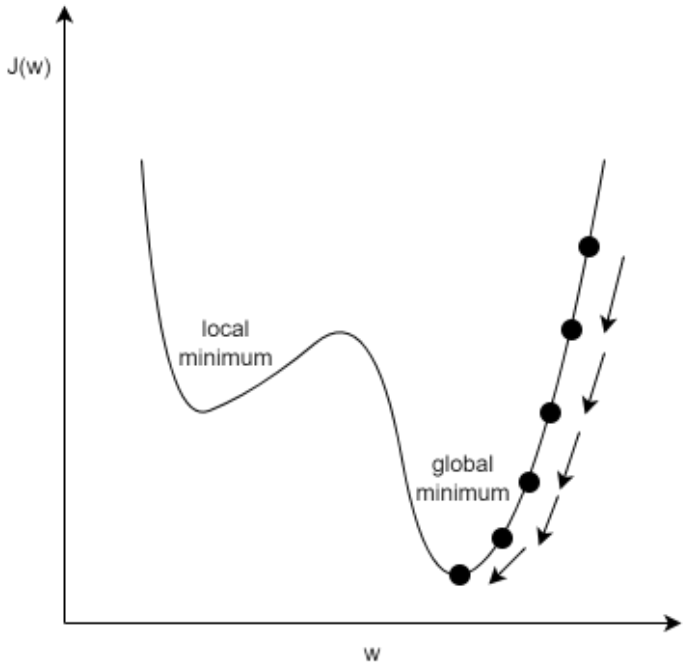

Figure (2.3) Simplified representation of the cost function with gradient descent.

Since we are dealing with functions, it is possible to reach a minimum value. A minimum value of a loss function is basically the best generalization for a given sample of data. The way neural networks reach those minimum values is by using gradient descent. However, since the functions have a very large number of parameters, there are many local minimum values in which the algorithm may converge. Reaching the global minimum value is very hard in many situations, and a local minimum may yield satisfactory results. In a nutshell, the learning process may be seen as achieving the goal of reaching a minima of a determined cost function ($min(J(w))$), in which backpropagation is used to change the values of weights and biases from the last layers to the first layers in order to keep minimizing the cost function.

2.1.3 Convolutional Neural Networks

CNNs were game-changing in the image processing scenario. In many Kaggle competitions among tabular data, traditional ML methods such as Random Forests (Breiman, 2001), XGBoost (Chen et al., 2015), and SVM sometimes achieve even better results than Artificial Neural Networks (Drew and Monson, 2000). This is not the case for images, in which the disparity between traditional methods and CNN is very apart from each other.

Suppose a scenario in which we want to distinguish between cats and dogs. In some sense, they have many similarities. Nonetheless, cats have pointier ears than dogs. The pointier ears are a shape and require a group of pixels in a particular order to understand this pattern. This is where CNN's are so powerful. Instead of the features being related to a single pixel, it extracts information in various levels of abstraction, which tends to be at a higher level in the last layers.

A typical CNN structure is composed of three types of layers (Albawi et al., 2017): (1) convolutional layer, (2) pooling layer, and (3) fully-connected layer. The convolution operation requires a filter (also refered to as a kernel) and the original image which is as array with three dimensions (height, width, and channels). The filter will then pass through the entire image and the convolution between the original array with the kernel will result in an array with new values. Figure 2.4 shows a simplified example of the convolution operation. To show exactly the mathematics behind it, we painted the pixels in red that ends up in the upper-left quadrant of the final matrix, being the sum of the multiplications at each position ($0*0+1*1+3*2+4*3 = 19$).

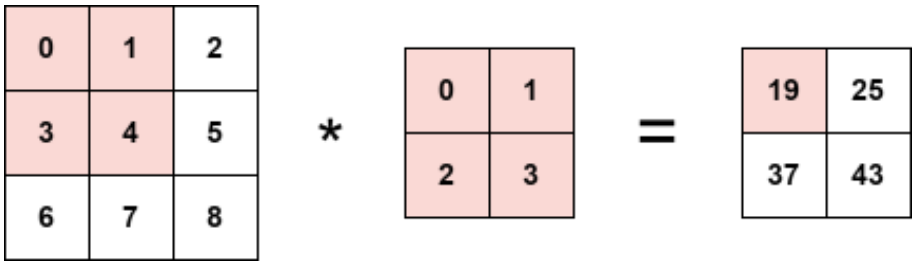

Figure (2.4) Example of the convolution operation.

Note that the output array is smaller than the input array in this case. The output size is affected by three factors: the size of the input image, the size of the kernel, and the stride value (how many pixels the kernel skips from one iteration to the other). In this case, every time we apply a convolution, the size of the image will be smaller, and often times this is not desirable. A common approach for maintaining the array dimensions is to apply padding (insert a border in the image with zeros). In this regard, considering the image height (H), width (W), kernel size (KxK), stride (S), and padding (P), the output dimensions of each convolution can be expressed by:

$$(\frac{H - K + 2P}{S} + 1, \frac{W - K + 2P}{S} + 1) \tag{2.2}$$

Just like in the neural networks, we also apply activation functions in the output of the convolutions. In CNNs, the Rectified Linear Units (ReLU) are the most used because they are

fast and avoid vanishing gradient problems. In this way, we can connect multiple convolutional layers.

Moving forward, we also have the pooling layer, which reduces the spatial dimensions of the arrays, reducing the number of parameters and computational cost of the CNN. There are two main pooling approaches, max pooling and average pooling. Figure 2.5 shows an example of the max and average pooling operations using a 2x2 kernel, in which the max pooling extracts the highest value and the average pooling extracts the average value, reducing the image dimensions.

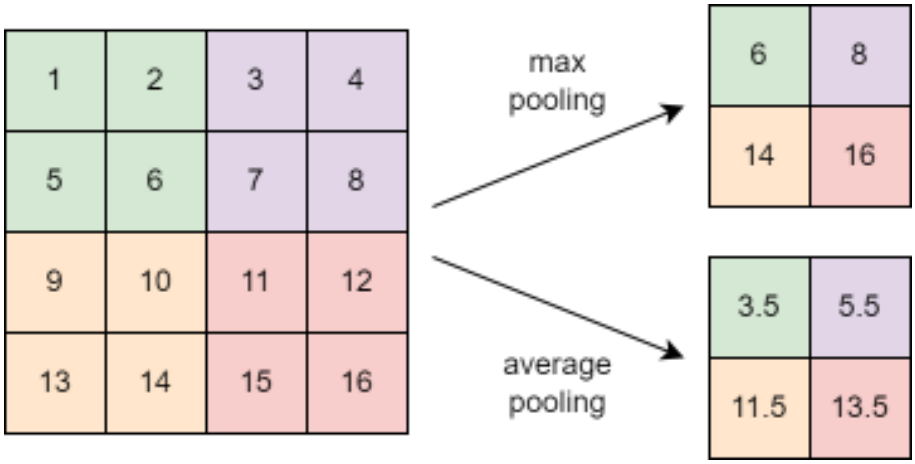

Figure (2.5) Example of the max and average pooling operations.

Finally, we connect the last layer with a fully connected layer to achieve a classification result. In binary classification problems, we only need a single neuron associated with a sigmoid activation function to present values in the range of zero and one. In cases with more classes, each of the last neurons will be responsible for holding the result of a single class. Now, there are two distinct tasks, multilabel (Tsoumakas and Katakis, 2007) and multiclass (Aly, 2005). In multiclass problems, there can only be one correct class. Thus, we use the softmax activation function, which makes the sum of all neurons' probabilities equal one. The multilabel still uses a sigmoid activation function, but now the classes do not sum up to one since all classes may be present simultaneously. A good example of this would be to feed pictures of humans and try to simultaneously predict age, gender, and ethnicity, among others.

2.1.4 Image segmentation

Image segmentation is referred to as the process of grouping the pixels of a given image into different groups, enabling a more synthetic understanding of the image. To group pixels, we first need to understand the two main categories of targets adopted by the computer vision community: "things" and "stuff" (Cordts et al., 2016; Everingham et al., 2015; Geiger et al., 2013; Lin et al., 2014; Neuhold et al., 2017). The thing categories are often countable objects and present characteristic shapes, similar sizes, and identifiable parts (e.g., buildings, houses, swimming pools). Oppositely, stuff categories are usually not countable and amorphous (e.g., lake, grass, roads) (Caesar et al., 2018).

Understanding the target type, the specific task to tackle the problem may be chosen. In this regards, there are three main approaches for image segmentation (Hoeser et al., 2020; Ma et al., 2019; Voulodimos et al., 2018; Yuan et al., 2021) (Figure 2.6): (1) semantic segmentation; (2) instance segmentation; and (3) panoptic segmentation. For a given input image (Figure 2.6A), semantic segmentation models perform a pixel-wise classification (Singh and Rani, 2020) (2.6B), in which all elements belonging to the same class receive the same label. However, this method presents limitations for recognising individual elements, especially in crowded areas. On the other hand, instance segmentation generates bounding boxes (i.e., a set of four coordinates that delimits the objects boundaries) and performs a binary segmentation mask for each element, enabling a distinct identification (He et al., 2020). Instance segmentation approaches are restricted to objects (Figure 2.6B), not covering background elements (e.g., lake, grass, roads). The panoptic segmentation task (Kirillov et al., 2019) aims to simultaneously combine instance and semantic predictions for classifying things and stuff categories, providing a more informative scene understanding (Figure 2.6D).

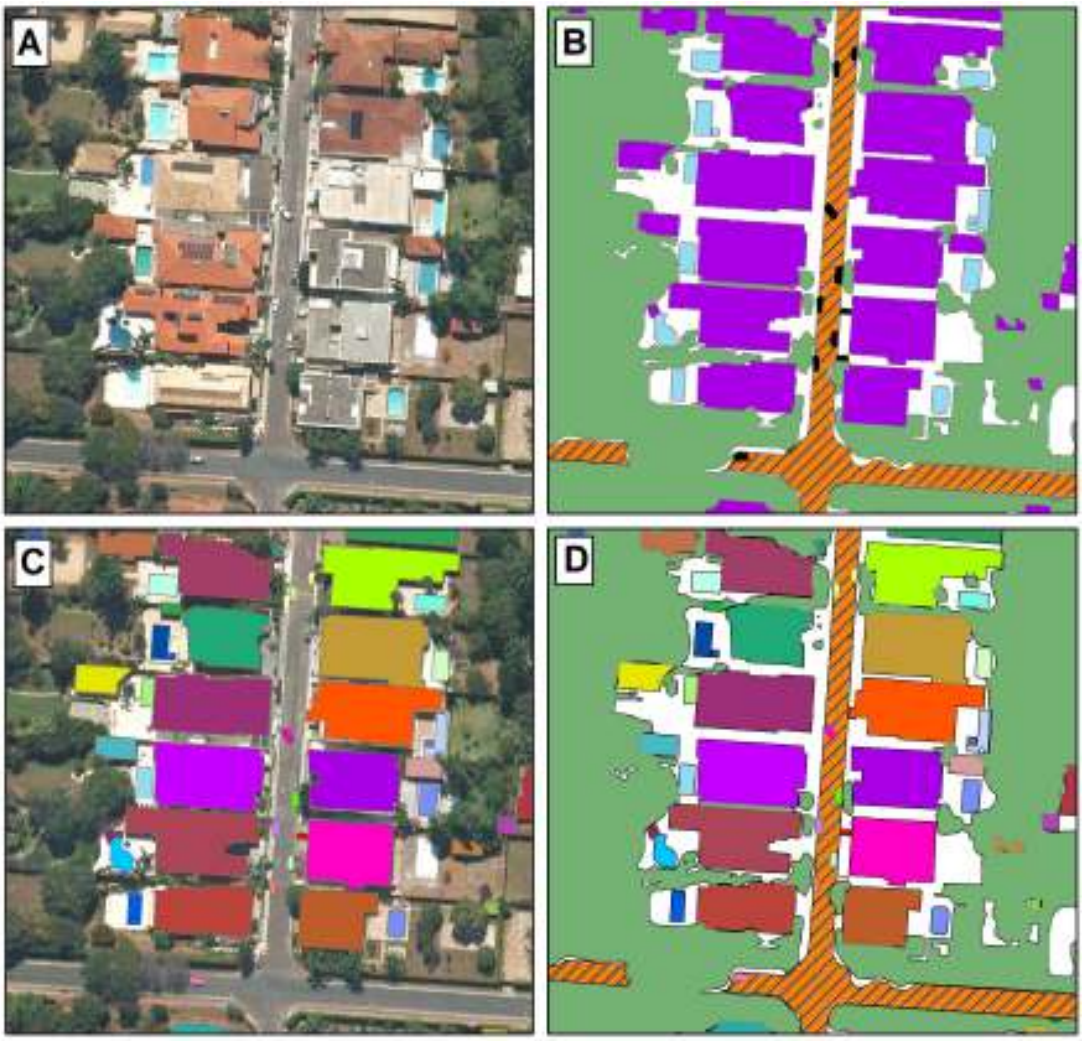

Figure (2.6) Representation of the (A) Original image, (B) semantic segmentation, (C) instance segmentation, and (D) panoptic segmentation.

Anchor and Bounding Box-Free approach

The conventional approaches, for instance and panoptic segmentation, are extensions of object detection networks that aim to predict a bounding box to each element. In general, those approaches require many parameters such as anchor boxes, anchor ratios, and regression losses for optimizing the bounding boxes, among others (Bonde et al., 2020; de Carvalho et al., 2021a). All of those parameters are learning-based strategies, and even though it has a high performance, removing some of those elements may be more practical. This field has two main categories: anchor-free and bounding-box-free methods. The anchor-free methods still yield bounding boxes at the end but eliminate the necessity of having anchor boxes in the training processes (Lee and Park, 2020; Xie et al., 2020). The second approach is a completely bounding box-free approach, which does not provide any boxes in the final results (Bonde et al., 2020; de Carvalho et al., 2021a).

Chapter 3

Box-Based Instance Segmentation

This chapter addresses box-based instance segmentation methods with two significant challenges for image processing, the small objects, and the usage of sliding windows for large scene classification. The analyzed target are the straw beach umbrellas (SBU), which are very common in the beach landscape. The results from this chapter were published in the ISPRS International Journal of Geoinformation.

3.1 Presentation

Public land management is essential for the effective use of natural resources with implications for economic, social, and environmental issues (Brown et al., 2014). Government policies establish public areas in ecological, social, or safety-relevant regions (i.e., natural fields and historic spaces), offering services ranging from natural protection to recreation (Brown et al., 2014; DeFries et al., 2007). However, managing public interests to promote social welfare over private goals is a significant challenge. Especially in developing countries, recurrent misuse of public land (Belal and Moghanm, 2011), and illegal invasions (i.e., the use of public lands for private interests) (Dacey et al., 2013) are among the most common problems.

Coastal zone areas concentrate a large part of the world population, despite being environmentally sensitive with intense natural processes (erosion, accretion, and natural disasters) (Brown et al., 2015) and constant anthropic threats (marine litter, pollution, and inappropriate use) (Martin et al., 2018; Serra-Gonçalves et al., 2019). The coastal zone is a priority for developing continuous monitoring and misuse detection programs. In Brazil, coastal areas belong to the Federal Government, considering the distance of 33 meters from the high-medium water line in 1831 (known as navy land). According to the Brazilian Forest Code, beaches and water bodies have guaranteed public access. Therefore, Brazilian legislation establishes measures for public use, economic exploitation, environmental preservation, and recovery considering coastal areas socio-environmental function. Inspecting beach areas in Brazil is challenging, as the Unions Heritage Secretariat does not have complete and accurate information about this illegal occupation throughout the country. The undue economic exploitation of the urban beach strip leads to an increase in the number of illegal constructions and a reduction in government revenue due to

non-registration, environmental problems, visual pollution, beach litter, among others. Many illegal installations on urban beaches are masonry constructions for private or commercial use. In addition, tourist infrastructure for food and leisure extends several straw beach umbrellas (SBUs) (fixed in the sand by local traders) to the sand strip without permission. Given the potential impact on the environment and the local economy, the monitoring and enforcement to curb private business development in public spaces must be constant and efficient (Brown et al., 2015), mainly to avoid uncontrolled tourism development (Burak et al., 2004; Gladstone et al., 2013). The inspection must ensure the legal requirements, avoid frequent changes that lead to lawful gaps, and minimize differences arising from conflicts of interest.

Conventionally, the inspection process imposes a heavy burden on state and federal agencies, containing few inspectors with low frequency on site. In this regard, geospatial technologies and remote sensing techniques are valuable for public managers since they enable monitoring changes in the landscapes and understanding different patterns and behaviors. An excellent potential for remote sensing application by government control agencies is detecting unauthorized constructions in urban areas (He et al., 2019b; Varol et al., 2019). Several review articles address the use of remote sensing and geospatial technology in coastal studies (El Mahrad et al., 2020; Lira and Taborda, 2014; McCarthy et al., 2017; Ouellette and Getinet, 2016; Parthasarathy and Deka, 2019). Currently, geospatial technology is a key factor for the development and implementation of integrated coastal management, allowing a spatial analysis for studies of environmental vulnerability, landform change (erosion and accretion), disaster management, protected areas, ecosystem, economic and risk assessment (Ibarra-Marinas et al., 2021; Poompavai and Ramalingam, 2013; Rifat and Liu, 2020; Sahana et al., 2019).

However, few remote sensing studies focus on detecting tourist infrastructure objects on the beach for inspection. Beach inspection requires high-resolution images and digital image processing algorithms that identify, count, and segment small objects of interest, such as the SBUs. Among the remote sensing data, high-resolution orbital images have the advantage of periodic availability and coverage of large areas at a moderate cost, unlike aerial photographs and Unmanned Aircraft Systems (UASs) of limited accessibility. Typically, high-resolution satellite images acquire a panchromatic band (from 1 meter to sub-metric resolutions) and multispectral bands (spectral bands of blue, green, red, and near-infrared with spatial resolutions ranging from 1m to 4m), such as IKONOS (Pan-chromatic: 1 m; Multispectral: 4 m), OrbView-3 (Panchromatic: 1 m; Multispectral: 4 m), QuickBird (Panchromatic: 0.6 m; Multispectral: 2.4 m), GeoEye-1 (Panchromatic: 0, 41 m; Multispectral: 1.65 m) and Pleiades (Panchromatic: 0.5 m; Multispectral: 2 m). Unlike the satellites mentioned above, the WorldView-2 (WV2) and WorldView-3 (WV3) images pre-sent a differential for combining the panchromatic band (0.3 m resolution) with eight multispectral bands (Resolution 1, 24 m): coastal (400 - 450 nm), blue (450 - 510 nm), green (510 - 580 nm), yellow (585 - 625 nm), red (655 - 690 nm), red edge (705 - 745 nm), near-infrared 1 (NIR1) (770 - 895 nm) and near-infrared 2 (NIR2) (860 - 1040 nm). There-fore, WorldView-2 and WorldView-3 have additional spectral bands compared to other sensors (coastal, yellow, red edge, and NIR2), valuable for urban mapping (Momeni et al., 2016). Therefore, the conjunction of the spectral and spatial properties of the WorldView-2

and WorldView-3 images is an advantage in the detailed classification process in complex urban environments. Few studies assess infrastructure detection on the beach. Llausàs et al. (2019) conducted private swimming pools on the Catalan coast to estimate water use from WorldView-2 images and Geographic Object-Based Image Analysis (GEOBIA). Papakonstantinou et al. (2019) used UAS images and GEOBIA to detect tourist structures in the coastal region of Santorini and Lesvos islands. Despite the wide use of the GEOBIA, Deep Learning (DL) segmentation techniques demonstrate greater efficiency than GEOBIA in the following factors: (a) greater precision and efficiency; (b) high ability to transfer knowledge to other environments and different attributes of objects (light, color, size, shape, and background); (c) requires less human supervision; and (d) less noise interference (Albuquerque et al., 2021a; Guirado et al., 2017; Huang et al., 2020; Liu et al., 2018c).

The DL methods promote a revolution in several fields of science, including visual recognition (Voulodimos et al., 2018), natural language processing (Sun et al., 2017; Young et al., 2018), speech recognition (Nassif et al., 2019; Zhang et al., 2018c) object detection (Sharma and Mir, 2020; Zhao et al., 2019) medical image analysis (Liu et al., 2019b; Serte et al., 2020; Zhou et al., 2019), person identification (Bharathi and Shamily, 2020; Kaur et al., 2020), among others. Like other fields of knowledge, DL achieves state-of-the-art performance in remote sensing (Li et al., 2020a; Liu et al., 2020; Ma et al., 2019) with a significant increase in articles after 2014 (Cheng et al., 2020b). In a short period, several review articles have reported about DL and sensing, considering different applications (Ball et al., 2017; Li et al., 2019a); digital image processing methods (Cheng et al., 2020b; Hoeser et al., 2020; Khelifi and Mignotte, 2020; Li et al., 2018; Ma et al., 2019; Zhang et al., 2016); types of images (Paoletti et al., 2019; Parikh et al., 2020; Signoroni et al., 2019; Vali et al., 2020; Zhu et al., 2017), and environmental studies (Yuan et al., 2020). DL algorithms use neural networks (Schmidhuber, 2015), a structure composed of weighted connections between neurons that iteratively learn high and low-level features such as textures and shapes through gradient descent (Nogueira et al., 2017). CNN have great usability in image processing because of their ability to process data in multi-dimensional arrays (Lecun et al., 2015). There are many applications with CNN models, e.g., classification, object detection, semantic segmentation, instance segmentation, among others (Ma et al., 2019). The best method often depends on the problem specification.

Instance segmentation and object detection networks enable a distinct identification for elements of the same class, suitable for multi-object identification and counting. A drawback when comparing instance segmentation and object detection networks is real-time processing, in which instance segmentation usually presents an inference speed lower than object detection. Nevertheless, instance segmentation models bring more pixel-wise information, crucial to determining the exact object dimensions.

However, instance segmentation brings difficulties in its implementation. The first is the annotation format, where most instance segmentation models use a specific format that is not as straightforward as traditional annotations. The second is that most algorithm uses conventional Red, Green, and Blue (RGB) images, whereas remote sensing images often present more spectral channels and varied dimensions. The third problem is adjusting the training images to

a specific size to train the models. To classify a large area requires post-processing procedures. Object detection algorithms require only the bounding box coordinates, which are much more straightforward than instance segmentation, which requires each object's bounding boxes and polygons.

Another recurrent problem is the poor performance of DL algorithms on small objects since they present low resolutions and a noisy representation (Li et al., 2017). Common Objects in Context (COCO) (Lin et al., 2014) characterizes objects sizes within three categories: (a) small objects (area $< 32^2$ pixels); (b) medium objects ($32^2 <$ area $< 96^2$); and (c) large objects (area $> 96^2$ pixels). The average precision (AP) score (main metric) has nearly half of the performance on small objects within the COCO challenge than on medium and large objects. According to a review article by Tong et al. (2020), few studies focus on small object detection, and despite the subject's relevance, the current state is far from acceptable in most scenarios and still underrepresented in the remote sensing field. In this regard, an effective method is to increase the image dimensions. In this way, the small objects will have more pixels, differentiating them from noise.

The present research aims to effectively identify, count, and estimate SBU areas using multispectral WordView-3 (WV-3) imagery and instance segmentation to properly inspect and properly control tourist infrastructure. Very few works use instance segmentation in remote sensing, and none use WV-3 images or in beach areas. Our contributions are threefold: (1) a novel application of instance segmentation using multispectral WV-3 images on beach areas, (2) leverage the existing method for classifying large areas using instance segmentation, and (3) analyze and compare the effect of the DL image tiles and their metrics.

3.2 Materials and Methods

The methodology is subdivided into the following steps: (A) dataset; (B) instance segmentation approach; (C) image mosaicking using sliding window; and (D) performance metrics (Figure 3.1).

3.2.1 Dataset

Study Area

The study area was Praia do Futuro in Fortaleza, Ceará, Brazil, with intense tourist activities. The research used WorldView-3 images from September 17, 2017, and September 18, 2018, provided by the European Space Agency (ESA) with a total length of 400km^2 (Figure 3.2. The WorldView-3 images combine the acquisition of panchromatic (with 0.31-m resolution) and multispectral (with 1.2-m resolution and eight spectral bands) bands. We used the Gram-Schmidt pan-sharpening method (Laben and Brower, 2000) with nearest neighbor resampling to maximize image resolution and preserve spectral values (Johansen et al., 2020). The pansharpening technique aims to combine the multispectral images (with low spatial resolution and narrow spectral band) with the panchromatic image (with high spatial resolution and wide spectral

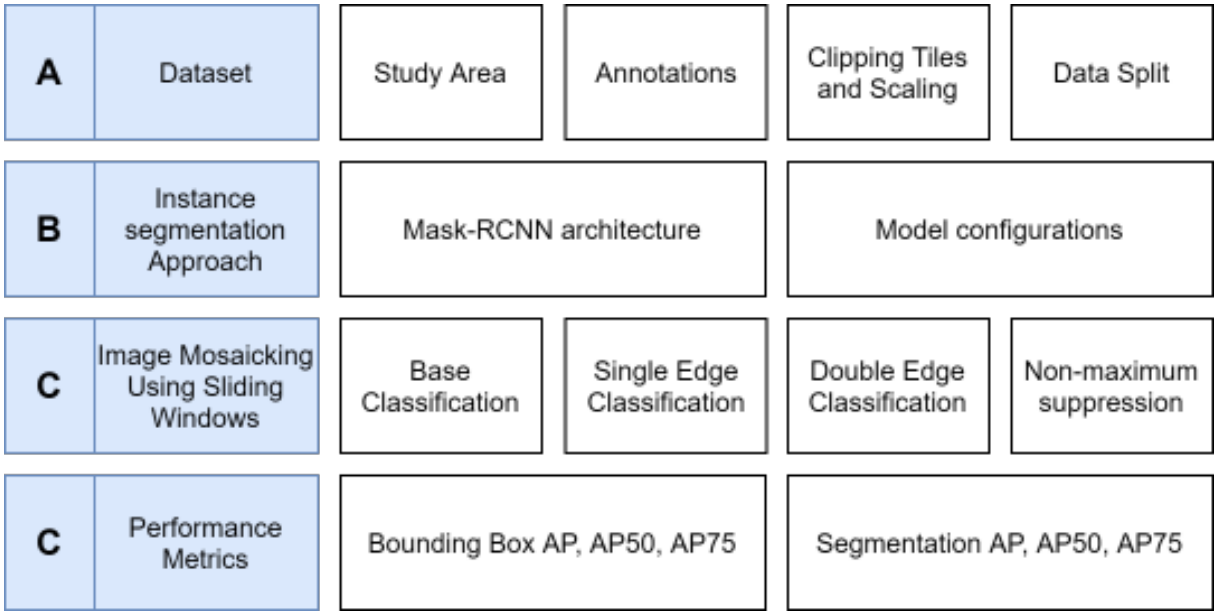

Figure (3.1) Methodological flowchart.

band), extracting the best characteristics of both data and merging in a product that favours the data interpretation (Ghassemian, 2016). The Gram-Schmidt technique presents high fidelity in rendering spatial features, being a fast and straightforward method.

Annotations

Annotations assign specific labels to the objects of interest, consisting of the ground truth in model training. Instance segmentation programs use the COCO annotation format, such as Detectron2 software (Wu et al., 2019) with the Mask-RCNN model (He et al., 2017). Consequently, several annotation tools have been proposed for traditional photographic images considering the COCO format, such as LabelMe (Russell et al., 2008; Torralba et al., 2010), Computer Vision Annotation Tool (CVAT) (Sekachev et al., 2019), RectLabel (`https://rectlabel.com`), Labelbox (`https:///labelbox.com`), and Visual Object Tagging Tool (VoTT) (`https://github.com/microsoft/VoTT`). In remote sensing studies, an extensive collection of annotation tools is present in Geographic Information Systems (GIS) with several procedures to capture, store, edit and display geo-referenced data. Therefore, an alternative to taking advantage of all the technology developed for spatial data is to convert the output data from the GIS program to the COCO annotation format. In the present research, we converted GIS data to the COCO annotation format from the program developed in the C++ language proposed by Carvalho et al. (2021). The SBUs' ground truth digitization used ArcGIS software. Since instance segmentation requires a unique identifier (ID) for each object, each SBU had a different value (from 1 to N, with N being the total number of SBUs).

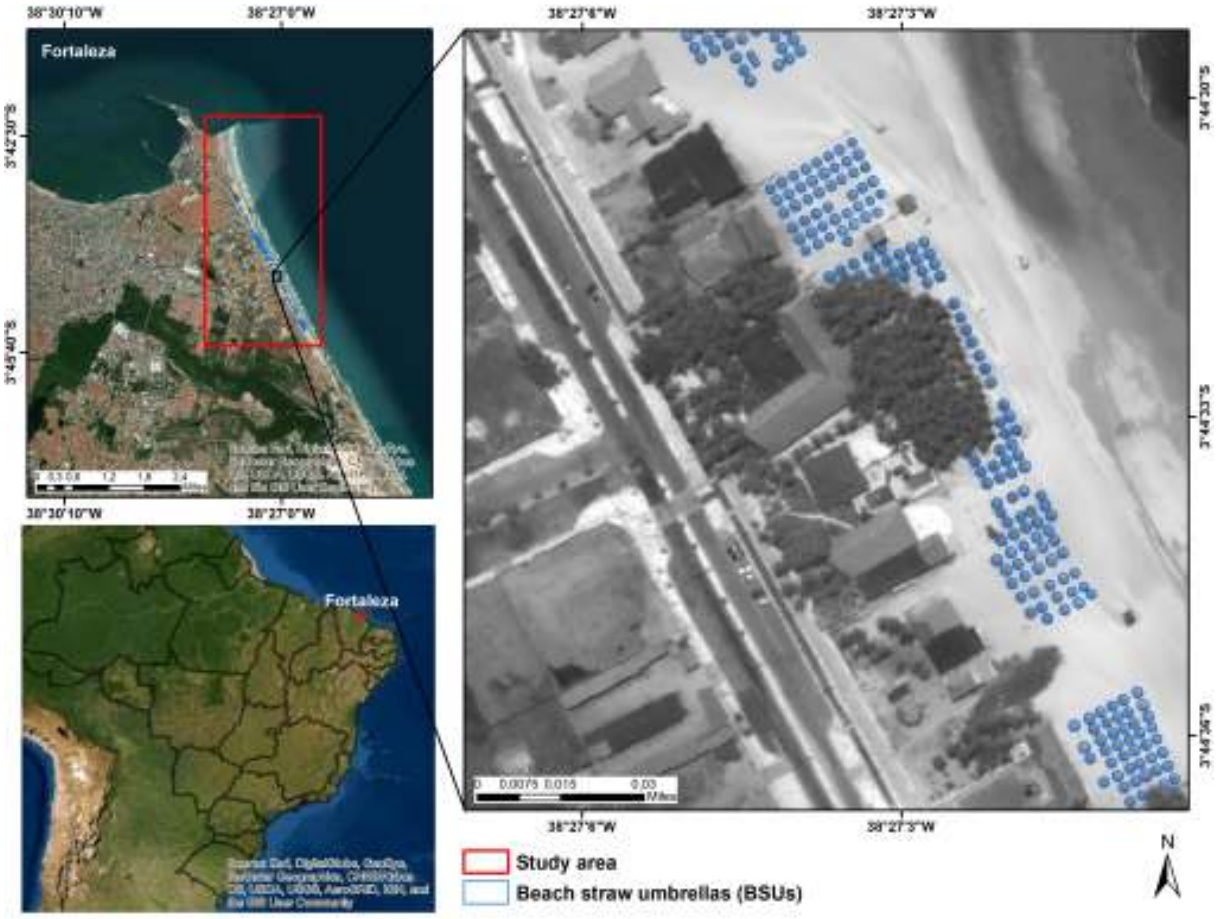

Figure (3.2) Study Area with a zoomed area from the WorldView-3 image.

Clipping Tiles and Scaling

Our research targets are very small ($< 16^2$ pixels) and very crowded in most cases. To improve small objects' detection, a powerful yet straightforward operation is to scale the input image (Tong et al., 2020). We evaluated the ratios of 2x, 4x, and 8x in the original image. The cropped tiles considered 64x64 pixels in the original image, which increased proportionally with the different scaling ratios (128x128, 256x256, and 512x512, respectively).

Data Split

For supervised DL tasks, using three sets is beneficial to evaluate the proposed model. The training set usually presents most of the samples, which is where the algorithm will understand the patterns. However, the training set alone is insufficient since the final model may be overfitting or underfitting. In this regard, the validation set plays a crucial role in keeping track of the model progress. A common approach is to save the model with the best performance on the validation set. Nevertheless, this procedure also brings a bias. With that said, the model is often preferable to be done using an independent test set. We distributed the cropped tiles into training, validation, and test sets as listed in Table 3.1. The number of instances shows a high object concentration, with an average of nearly ten objects per 64x64 pixel image.

Table (3.1) Data split between the training validation and testing sets with their respective number of images and instances.

Set	Number of images	Number of instances
Train	185	1,780
Validation	40	631
Test	45	780

3.2.2 Instance Segmentation Approach

Mask-RCNN Architecture

Facebook Artificial Intelligence Research (FAIR) introduced the Mask-Region-based Convolutional Neural Network (Mask-RCNN) as an extension of previous studies for object detection architectures: RCNN (Girshick et al., 2014), Fast-RCNN (Girshick, 2015), and Faster-RCNN (Ren et al., 2017). The Mask-RCNN uses the Faster-RCNN as a basis with the addition of a segmentation branch that performs a binary segmentation on each detected bounding box using a Fully Convolutional Network (FCN) (Shelhamer et al., 2017) (Figure 3.3).

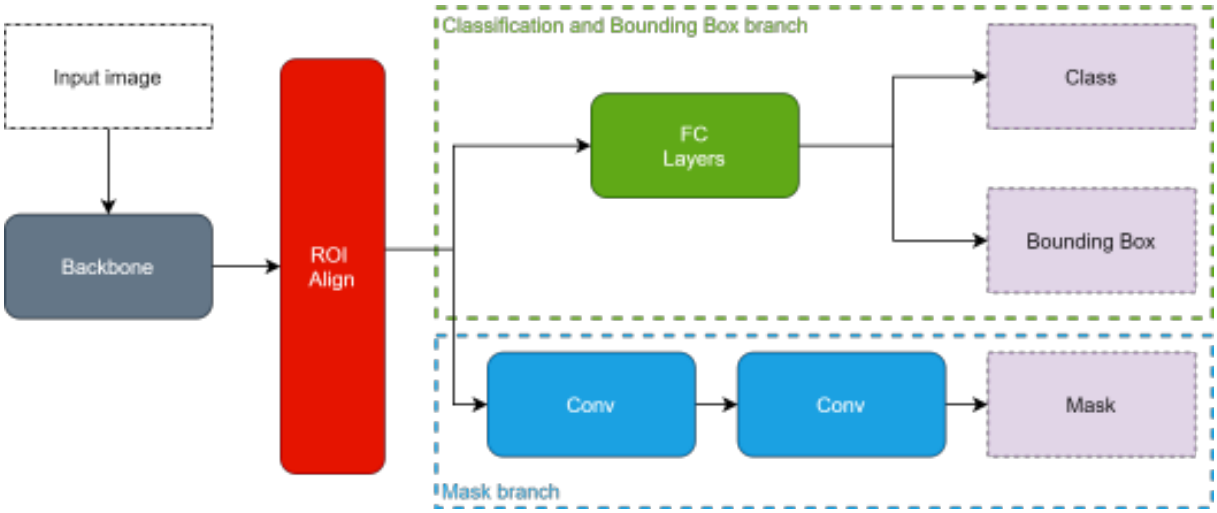

Figure (3.3) Mask-RCNN Architecture.

The region-based algorithms present a backbone structure (e.g., ResNets (He et al., 2016), ResNeXts (Xie et al., 2017), or other CNNs) followed by a Region Proposal Network (RPN). However, the Mask-RCNN has a Region of Interest (RoI) Align mechanism, in contrast to the RoIPool. The benefit of this method is a better alignment of each RoI with the inputs that removes any quantization problems on the RoIs boundaries. Succinctly, the model aims to identify the bounding boxes, classify the bounding box classes, and apply a pixel-wise mask on the bounding box objects. The loss function considers the three elements, being the sum of the bounding box loss ($Loss_{bbox}$), mask loss ($Loss_{mask}$), and classification loss ($Loss_{class}$), in which Lossmask and $Loss_{class}$ are log loss functions, and $Loss_{bbox}$ is the L1 loss.

We use the Detectron2 software (Wu et al., 2019), which uses the Pytorch framework. Since this architecture is usually applied to traditional images (3 channels), it requires some adjust-

ments to be compatible with the WV-3 imagery (TIFF format and has more than three channels) (Carvalho et al., 2021).

Model Configurations

To train the Mask-RCNN model, we made the necessary source code changes for compatibility and applied z-score normalization based on the training set images. We only used the ResNeXt-101-FPN (X-101-FPN) backbone since the objective is to analyze scaling.

Regarding hyperparameters, we applied: (a) Stochastic gradient descent (SGD) optimizer with a learning rate of 0.001 (divided by ten after 500 iterations); (b) 256 ROIs per image; (c) five thousand iterations; (d) different anchor box scales for the original image (4, 8, 12, 16, 32), 2x scale image (8, 16, 24, 32, 64), 4x scale image (16, 32, 48, 64, 128), and 8x scale image (32, 64, 48, 128, 256). To avoid overfitting, we applied the following augmentation to the training images: (a) random horizontal flip, (b) random vertical flip, and (c) random rotation. Finally, we used Nvidia GeForce RTX 2080 TI GPU with 11GB memory to process and train the model.

3.2.3 Image Mosaicking Using Sliding Windows

In remote sensing, the images often present interest areas much larger than the images used in training, validation, and testing. This problem requires some post-processing procedures. This process is not straightforward since the edges of the frames usually present errors. In this context, the sliding window technique has been used for various semantic segmentation problems (Audebert et al., 2017a; da Costa et al., 2021b; de Albuquerque et al., 2020b), in which the authors establish a step value (usually less than the frame size) and take the average from the overlapping pixels to attenuate the border errors. The problem persists in object detection and instance segmentation since predictions from adjacent frames would output distinct partial predictions for the same object. Recently, Carvalho et al. (2021) proposed a mosaicking strategy for object detection using a base classifier (Figure 3.4B), vertical edge classifier (Figure 3.4C), and horizontal edge classifier (Figure 3.4E). Our research adapted the method by adding a double-edge classifier since some errors may persist (https://github.com/osmarluiz/Straw-Beach-Umbrella-Detection).

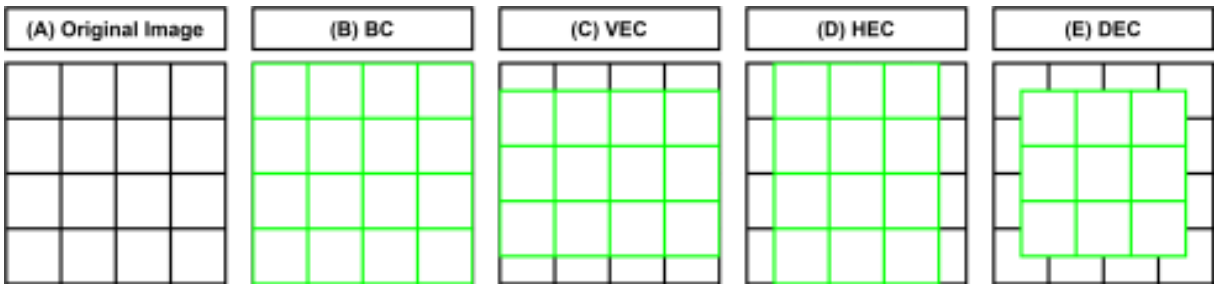

Figure (3.4) Scheme of the mosaicking procedure, with the Base Classification (BC), Vertical Edge Classification (VEC), Horizontal Edge Classification (HEC), and Double Edge Classification (DEC).

Base Classification

The first step is to apply a Base Classifier (BC) (considering all elements) using a sliding window starting at x=0 and y=0 and stride values of 512 (Figure 3.5B). This procedure produces partial classification on the frame's edges between consecutive frames, resulting in more than one imperfect classification for the same object, which is a misleading result.

Single Edge Classification

The second step is to classify objects located in the borders (partially classified objects by the BC). We applied the Vertical Edge Classifier (VEC) to classify elements in consecutive frames vertical-wise, composed of a sliding window that starts at x=256 and y=0 (Figure 3.5C). Similarly, to horizontal-wise consecutive frames, we applied the Horizontal Edge Classifier (HEC), with a sliding window that starts at x=0 and y=256 (Figure 3.5D). Both strategies use 512-pixel strides. In addition, to avoid the high computational cost, the VEC and HED only classify objects that start before the center of the image (x<256 for the VEC and y<256 for the HEC) and end after the image's center (x>256 for the VEC and y>256 for the HEC).

Double Edge Classification

An additional problem for crowded object areas such as SBUs is objects located at the BC borders horizontal-wise and vertical-wise, presenting a double edge error (DEC). Thus, we enhanced the mosaicking by applying a new sliding window, starting at x=256 and y=256 with 512-pixel strides (Figure 3.5E).

Non-maximum suppression sorted by area

Furthermore, each object located at the images' borders may present more than one classification for the same object, partial classifications from consecutive BC frames (incorrect classifications), and a unique, complete classification (correct classification) from the HEC, VEC, or DEC (Figure 3.5). The elimination of excessive boxes used the non-maximum suppression ordered by area, guaranteeing only the classification of the most significant element (complete object). Figure 3.5 shows an example of an element located at double edges, where the DEC classification is the largest and the correct one.

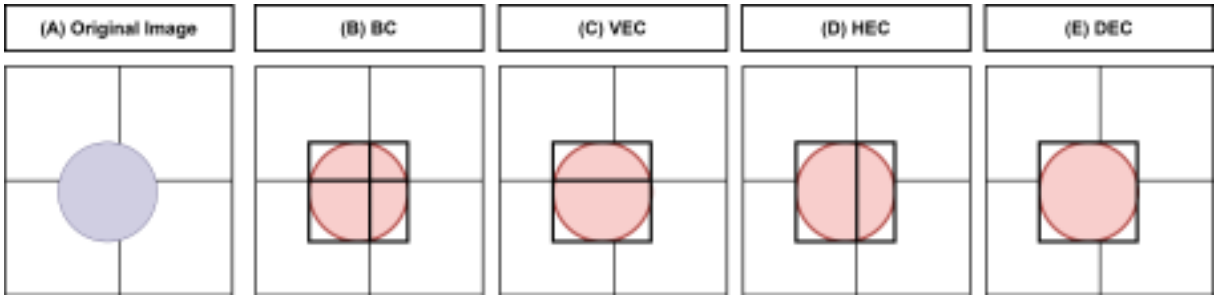

Figure (3.5) Example of classifications from each mosaicking procedure.

3.2.4 Performance Metrics

The model evaluation considered the following COCO metrics: Average Precision (AP), AP50, and AP75. The AP is a ranking metric that calculates the area under the precision-recall curve. However, in object detection, it is crucial to determine a minimum overlap between the predicted bounding box and the ground truth bounding box to evalu-ate a correct classification. Another element is the Intersection over Union (IoU) (Figure 3.6). In this regard, the COCO AP considers the average among ten Intersection over Union (IoU) thresholds (from 0.5 to 0.95 with 0.05 steps), while AP50 and AP75 scores consider a fixed threshold of 0.5 and 0.75.

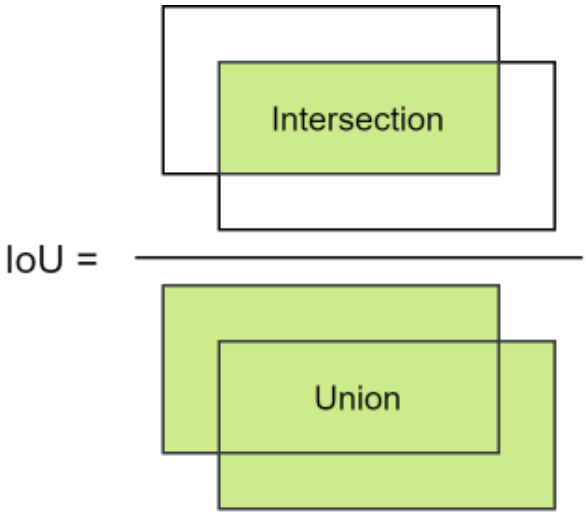

Figure (3.6) Representation of the Intersection over Union metric.

3.3 Results

3.3.1 Performance Metrics

Table 3.2 lists the detection (Box) and segmentation (Mask) results with different image scaling ratios and the X-101-FPN backbone. Results on the original image presented similar results compared to the COCO dataset scores. Scaling presented significant improvement, in which 2x scaling increased nearly 20% in the AP score, and 8x scaling increased nearly 30% AP.

Small objects negatively affect the strictest metrics (highest IoU, e.g., AP75). Slight errors in the bounding box position on small objects (with fewer pixels) significantly reduce the IoU (implying low AP scores). In turn, the mistakes are much less impactful when increasing the image dimensions. However, the high computational cost is a limitation of the indefinite increase in the image dimensions.

3.3.2 Scene Classification

We used the X-101-FPN model with the best scaling ratio (8x) scores, applying it in a 3072x2048 pixel image (also using 8x scaling) to validate the mosaicking technique. Figure 3.7A demonstrates a satisfactory classification even in crowded areas. This process excluded 66 partial

Table (3.2) COCO metrics (AP, AP50, and AP75) for segmentation (mask) and detection (box) on the different ratio images.

Ratio (Size)	Type	AP	AP50	AP75
8x (512x512)	Box	58.12	94.56	66.06
	Mask	56.76	93.73	63.86
4x (256x256)	Box	53.45	93.01	60.76
	Mask	52.89	92.21	58.87
2x (128x128)	Box	48.24	89.66	46.54
	Mask	49.09	90.24	49.84
1x (64x64)	Box	30.49	74.68	15.68
	Mask	36.69	77.42	27.50

classifications in total (Figure 3.7B), and the trained model has proven to distinguish SBUs from other elements such as tourist beach umbrellas.

Figure 3.8 shows three zoomed areas (1, 2, and 3) where the top images present the complete (correct) classification results, whereas the bottom images show the partial (incorrect) classifications deleted by the non-max suppression sorted by area algorithm. Figs. 3.8.1, 3.8.2, and 3.8.3 shows the DEC, VEC, and HEC, respectively. Another interesting point is that example 3.8.2 shows that one of the partial predictions has greater confidence than the correct prediction (97% against 96%), demonstrating that the non-maximum suppression ordered by area brings improved results.

Table 3.3 lists quantitative values that may be very helpful in decision-making. This methodology enables automatic counting and detection within large areas using Mask-RCNN. The sizes of the SBUs are very similar, with the average and median sizes very close and a standard deviation of $0.2m^2$. Besides, the algorithm could differentiate very close objects, showing a good usage of instance segmentation models for crowded regions.

Table (3.3) Analysis of the detected objects regarding their counting, average size, median size, minimum size, maximum size, and standard deviation, considering the 8x scaled image.

Description	Result
Count	148
Average SBU size	4,172 pixels (5.8m^2)
Median SBU size	4,027 pixels (5.6m^2)
SBU Standard Deviation	161.60 (0.2m^2)
Min. SBU size	2,693 pixels (3.8m^2)

3.4 Discussion

Instance segmentation is a state-of-the-art computer vision segmentation method that enables many practical approaches for identifying objects at the pixel level. Most instance segmentation studies use large datasets (e.g., COCO (Lin et al., 2014), Cityscapes (Cordts et al., 2016), Mapillary Vistas (Neuhold et al., 2017)) in a ready-to-use format. Developing datasets for instance segmentation is highly complex and labor-intensive, requiring annotation experts and a

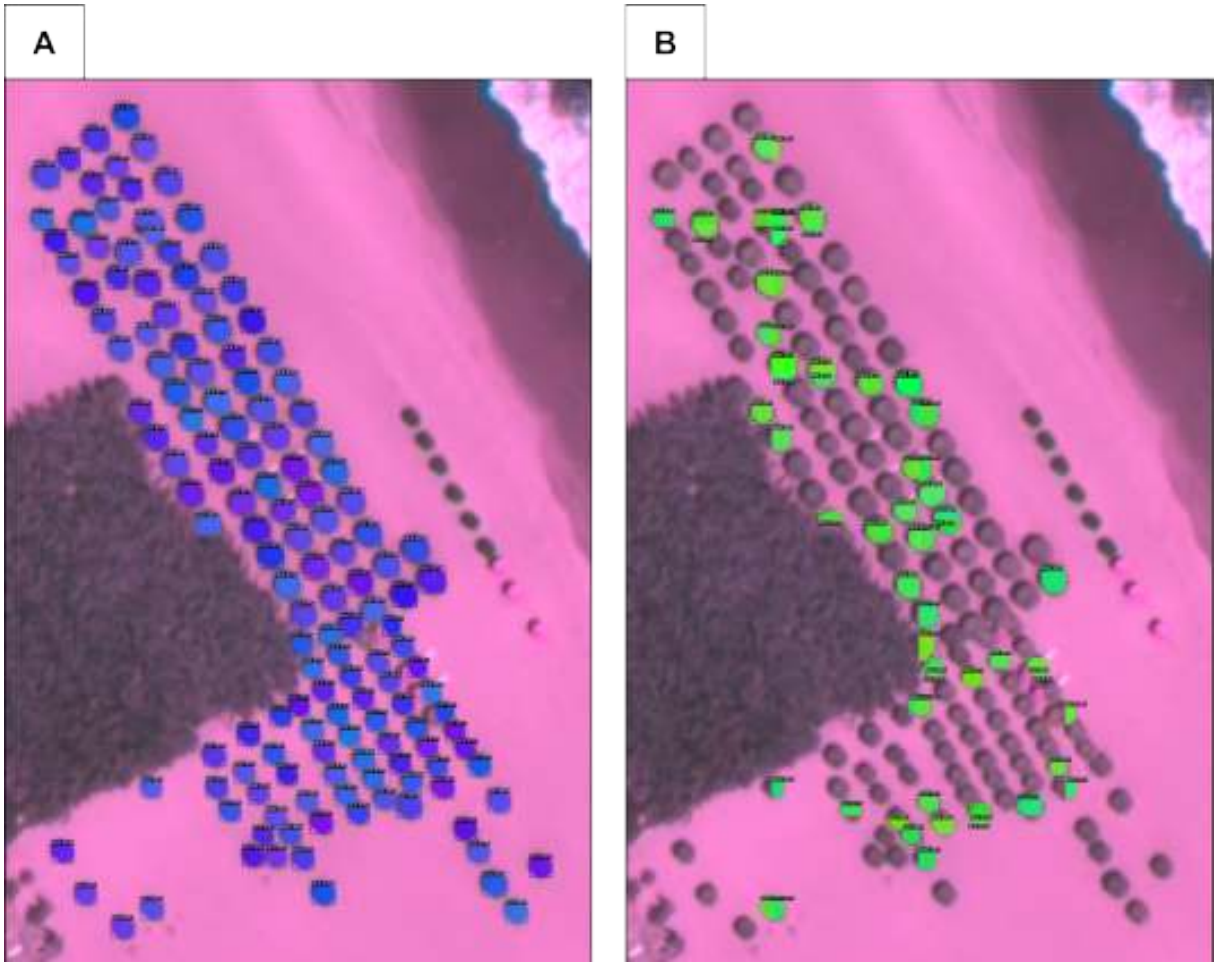

Figure (3.7) Classifications considering the correct classifications (A) and the deleted partial classifications from each object (B).

suitable storage format for DL models. Difficulties worsen for orbital remote sensing images by the need to choose the places of each image tile and the existence of very little annotation software available that considers geospatial data's particularities. With that said, in a Web of Science search up to November 11, considering the keywords instance segmentation, remote sensing, and deep learning, we found only 22 peer-reviewed journal articles. Despite the gains in efficiency and quality of results, the limited number of papers using instance segmentation demonstrates the difficulties reported. The present research demonstrates that instance segmentation allows a significant gain in inspection efficiency in coastal areas that have not yet been explored. Within these 22 articles, Soloy et al. (2020) also explored the beach areas, but with a different approach, as the authors used photos taken by the iPhone to quantify grain size on pebble beaches.

3.4.1 Multichannel Instance Segmentation Studies

Few studies addressed instance segmentation using multi-channel imagery. Most studies use RGB images (Li and Chen, 2021; Wu et al., 2021; Zhao et al., 2021) or even three-channel images from the combination of digital orthophoto map and near-infrared band from the Landsat-8 (Lv

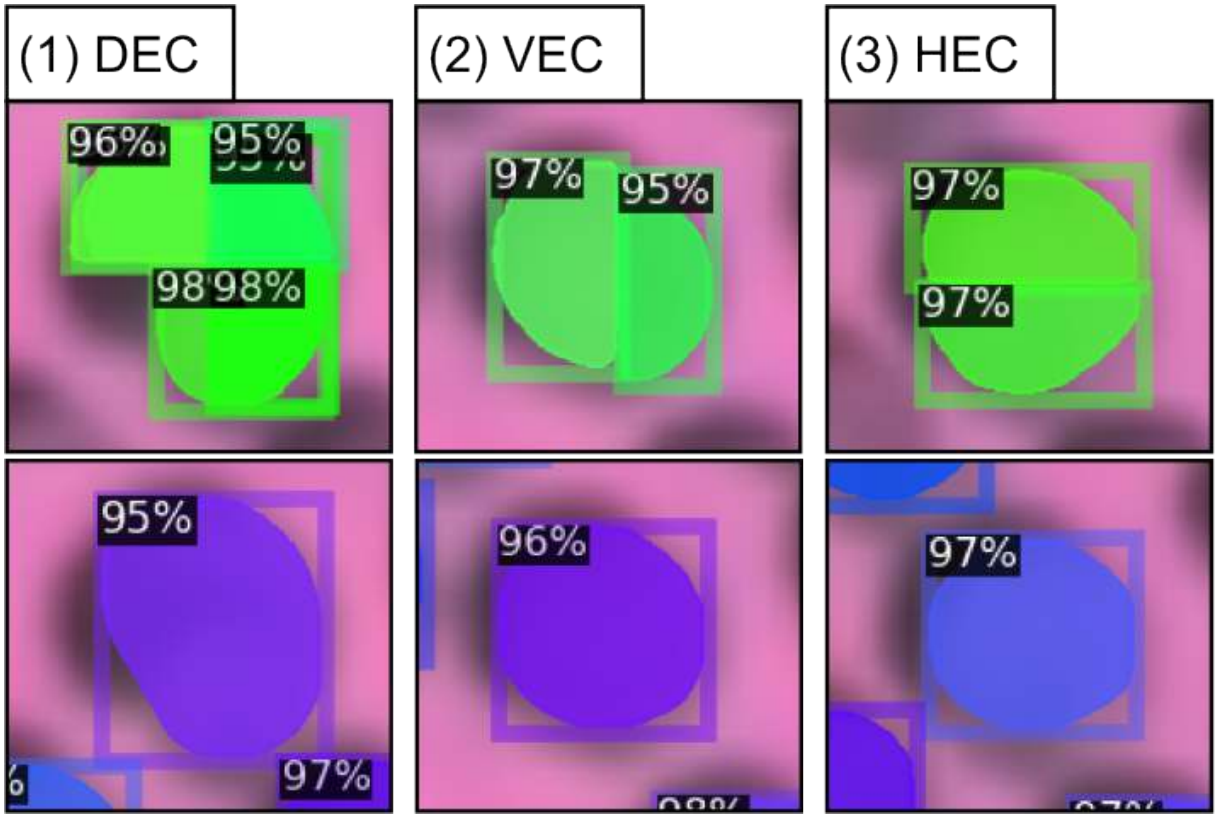

Figure (3.8) Representation of the three distinct classification scenarios, considering the double edge classification (DEC), vertical edge classification (VEC), and horizontal edge classification (HEC).

et al., 2020). The usage of multi-channels in remote sensing is widespread, allowing for more efficient detection than traditional RGB images (e.g., camera photos). Basically, there are four scenarios in remote sensing for using multi-channel inputs: (1) sensors with many spectral bands, (2) time series, (3) change detection, and (4) a combination of these characteristics (e.g., a time series of multispectral images). Using multispectral imagery, Carvalho et al. (2021) made a study on center pivot irrigation systems using Landsat-8 images. The authors compared the usage of seven channels with the traditional RGB, showing a difference of 3% in the AP metric when using more channels. Hao et al. (2021) used a multiband input for the Mask-RCNN for identifying tree crowns and estimating their height. Concerning time series applications, Albuquerque et al. (2021a) used Sentinel-1 time series (up to eleven channels) for mapping center pivots. The authors reported an increased performance when including more time frames. In a different approach, Albuquerque et al. (2021b) used Sentinel-2 time series (up to 24 channels), considering four spectral bands per temporal frame for effectively mapping regions with a cloud presence.

3.4.2 Methods for large scene classification

A significant problem is a DL adaptation for remote sensing applications that uses large-size images. In this regard, the present research used mosaicking with sliding windows for object

detection/instance segmentation. This procedure is more common in semantic segmentation approaches using overlapping pixels (Audebert et al., 2017a; Costa et al., 2021; da Costa et al., 2021b; de Albuquerque et al., 2020b). The method uses a sliding window with a step size smaller than the frame dimensions, causing overlapping. Averaging the overlapping areas mitigates errors, providing better accuracy metrics and visual results. However, for instance segmentation models, the procedure must consider the bounding boxes. In this sense, we modified the method proposed by Carvalho et al. (2021), introducing the double edge classifier (DEC) that is more efficient in extremely crowded areas, such as the SBUs. The methodology effectively eliminates frame discontinuity problems by considering the prediction under the best circumstance, providing a viable solution for mapping large areas.

Applying an instance segmentation algorithm over a large area enables a thorough scene understanding, which is vital for public inspection. For example, our study allows automatic counting of all SBUs and a series of other statistics, such as average size, median size, and standard deviation of the sizes. These quantitative results increase the amount of information for public managers to act, allowing the extraction of the exact location of each element by getting the coordinates of each bounding box.

3.4.3 Small object problem

Small objects often underperform in many datasets. For example, in the COCO dataset, the AP_{small} metric is much lower than the AP_{medium} and AP_{large} metrics. This effect is related to increasing noise with decreasing object size. In the review of Tong et al. (2020), image scaling is a straightforward approach to improve small object detection. Nevertheless, no study compares the effect of different scaling and improved object detection. In this regard, this research compares three scaling ratios for mapping SBUs, which are very small objects. This comparison can guide other studies further studies of small object detection in other scenarios. Our results show that image scaling (even as an image augmentation built-in method) may be a plausible and effective solution. The AP metrics increased more than 20%, considering eight times the original size. Even so, doubling the dimensions already provided a significant increase. This analysis is relevant since increasing the image dimensions might present computational problems (e.g., memory, processing time).

Some other alternatives have been studied for detecting small objects. Zhang et al. (2019) proposed a scale adaptive proposal network by modifying the Faster-RCNN architecture. This innovative approach has broad applications where there are datasets of many different sizes. Nonetheless, considering different scales might not be enough for very small objects, especially for AP scores, where few mistakes in the bounding box drastically reduce this accuracy metric. Generative Adversarial Networks (GAN) algorithms also present advances in studies with small objects (Li et al., 2017). In remote sensing, Ren et al. (2018) proposed an advanced end-to-end GAN to increase image resolution and apply the Faster-RCNN network in object detection. Therefore, a viable alternative for future studies would be the development of algorithms using GAN for surveillance in coastal areas. In the traditional RGB images from the COCO data set,

Kisantal et al. (2019) made an augmentation system based on copying and pasting small objects into different images to increase the representativeness of a small object in a larger number of images. This augmentation is a promising strategy for datasets with different scaled images. However, it can be computationally expensive in multichannel imaging and in detecting many small objects.

3.4.4 Accuracy metrics for small objects

Even though there is broad applicability of the COCO metrics for instance segmentation datasets (including the COCO dataset), the AP50 is the most appropriate metric for analyzing small objects (especially in datasets in which all objects are small) since very few mistakes drop the performance metrics significantly. Figure 9 shows two theoretical examples A and B, in which the prediction and the ground truth bounding boxes have the exact spatial dimensions.

When considering small objects, a slight mistake of 1 pixel horizontally and vertically has an IoU of 69.25%, impacting the AP and AP75 metric. A 1-pixel error in a 100x100 pixel bounding box generates a 96.10% IoU, showing the attenuation of slight errors in larger objects. This research shows that the simple increase in object dimensions gives the algorithm a better accuracy score. Therefore, generating ground truth data, especially for small objects, must be done rigorously to avoid misleading metrics.

3.4.5 Policy Implications

The Brazilian Government is responsible for the administration and inspection of federal properties. According to Normative Instruction No. 23, of March 18, 2020, the inspection action may have a preventive or coercive nature, requiring a field inspector to investigate possible irregularities committed against federal properties. The inspection action is predominantly coercive through denunciation, when the improper action is consolidated, leaving only the repair of the damage. The lack of preventive action causes an increase in unlawful acts and the filing of numerous lawsuits, with deprivation of use of areas and legal uncertainty.

In Brazil, beach areas are public properties protected by environmental legislation (CONAMA resolution No. 303 of 20/03/2002) as permanent preservation areas and consist of Navy land, where private occupation (private, commercial, or industrial) requires payment of a fee for the use of the public area. Beach areas are constant targets of economic exploitation and improper tourism and need constant surveillance. In this context, developing remote and semi-automated methods of surveillance of property misuse becomes fundamental.

Therefore, the instance segmentation of multispectral remote sensing images demonstrates a high potential to establish an effective action with a solid preventive impact due to the rapid infraction detection. However, the procedure should be improved, including other activities without prior authorization in coastal areas such as landfills, deforestation, construction, fences, or other improvements, which could be developed in future lines of research.

3.5 Conclusion

The automatic remote sensing detection of tourist infrastructure in beach areas is essential for government surveillance, requiring quick and periodic information for decision-making. The coastal regions of Brazil are government property, being areas with specific taxation for use and environmental protection. This study proposed a methodology based on instance segmentation to identify Straw Beach Umbrellas (SBUs), the most common tourist structure on Brazilian beaches. The developed methodological approach integrates different solutions for the use of instance segmentation in remote sensing data: (1) multi-channel models, (2) small object detection, and (3) classification of large areas. Therefore, we modified Detectron2's Mask-RCNN model to account for multi-channel image inputs in TIFF format, compared different scaling ratios on the original image and improved the existing method for classifying large images using the sliding window technique. Our results show that increasing image dimensions significantly improve the AP metric from 30% to 58%. In addition, the less strict metric (AP50) showed results from 74% to 94%. Image scaling is a computationally expensive solution, so we initially considered the original image dimensions of 64x64 pixels. Even though we evaluated up to 8 times the original dimensions (resulting in a 512x512 image), a two-times resizing already provides a significant increase. The research needs to define a trade-off between computational cost and the quality of predictions.

Another problem is the accumulation of errors on the frame edges, which intensify with overcrowded objects. Our innovative proposal to use double edge classification (DEG) solved the problem simply and efficiently. The architecture of all exposed methods is a suitable solution for accurately detecting small objects in large areas using multispectral data, providing insightful information for public managers. For example, statistical analysis of the SBUs on a 3,072x2,048 test image identified 148 objects with an average size of $5.8m^2$. The bounding box centroid establishes the exact geographic location. Future studies on this area will consider more beach elements, exploring objects and background elements, and other segmentation tasks such as panoptic segmentation.

Chapter 4

Box-Free Instance Segmentation with non-Touching Objects

This chapter introduces a novel procedure to obtain instance-level predictions from semantic segmentation models in targets that are always apart from each other, facilitating procedures such as extracting polygons from the objects.

4.1 Presentation

The great challenge for the Brazilian energy sector is to expand its energy production capacity while maintaining a high share of renewable sources in the energy mix. One of the most important factors is to guarantee its commitments to reduce greenhouse gas emissions (GHGs), established by the energy sector through the Intended Nationally Determined Contributions (INDC) (Lima et al., 2020) and in line with the Paris agreement ratified by Brazil in September 2016 (Tollefson, 2020). The primary source of Brazilian energy is hydroelectricity, which has been the primary geopolitical strategy for the energy sector since the 1960s. This development model has made the nation the most dependent on hydroelectric energy in the world. In fact, Brazil has a great advantage in having a hydro-energy base, a renewable, storable, and fundamental source for stability in meeting the countrys energy demand, especially since large plants are beneficial for regulating the demands at a reasonable time when the energy loads fluctuate.

However, most sites with hydropower potential have already been explored for energy generation, and the unexplored large-scale projects are mostly located in the Amazon region. This region imposes massive restrictions on constructing new hydropower plants in the country due to significant socio-economic and environmental impacts, compromising fragile ecosystems and entailing high costs in the long term (Jiang et al., 2018; Mayer et al., 2021). Recently, the Belo Monte project, with an installed capacity of 11.23 GW, illustrates the challenges of installing hydroelectric dams in the Amazon region. The project had a budget of US$ 13.1 billion and flooded an area greater than 7,000 km2, bringing challenges to mitigate environmental (Castro-Diaz et al., 2018; Gauthier et al., 2019; Gauthier and Moran, 2018; Runde et al., 2020) and

socio-economic impacts (Bro et al., 2018; Calvi et al., 2020), demanding efforts in terms of population resettlement (Mayer et al., 2021).

Ensuring energy security in the face of the country's economic growth and maintaining a portfolio of renewable sources leads to a redirection of investments, efforts, and priorities for the decentralization of renewable technologies with an increase in the reliability of the supply of the electrical system and risk reduction. In this scenario, solar and wind energy acquire prominence in this reduction in hydroelectric participation and sustain a mostly renewable share in the mix as it is currently (Ferraz de Andrade Santos et al., 2020). The hydroelectric source represented 83% of installed capacity at the beginning of the century, and the expectation is to reduce to 46% by 2031, according to the Brazilian government's Ten-Year Energy Plan (PDE-2031) and considering the most remarkable water scarcity recorded in 2021, the biggest within the last 90 years (Brasil et al., 2022). Thus, the contribution of hydroelectricity in the last decade has gradually decreased for these new alternatives that have reached the gigawatt-scale (Mendes and Sthel, 2018). Besides, relying on a single natural energy source brings security issues since these renewable energies are susceptible to climatic variations, with a possible need to activate thermoelectric plants to meet domestic demands (Hunt. et al., 2018). Several studies point out this problem and analyze moments of the recent energy crisis in the country (Melo et al., 2019; Mendes and Sthel, 2017, 2018; Reichert and Souza, 2021).

In addition, wind and solar energy allow a decentralized production closer to the consumer, and technological advances promote the constant reduction of generation costs, overcoming technical barriers and making these sources increasingly competitive due to economic gains and efficiency. Among the advantages of wind and solar energy systems are carbon-free energy sources with low environmental impact, potential to mitigate greenhouse gas emissions, low operating and maintenance costs, high availability, strengthening of the ends of the network, reduction of energy transmission losses, and increased in the overall efficiency of the electrical system (Sampaio and González, 2017).

According to the Brazilian National Electricity Agency (ANEEL) data from the beginning of 2022, the number of wind power plants in operation was 809, with a granted power of 21.5 GW and supervised power of 21.4 GW which represents 11.77% of the Brazilian electricity mix. The number reaches 1190 units with a granted power of 34.9 GW from the wind farms under construction and construction not started. In Brazil, the Northeast region is the most promising and favorable for wind energy conversion due to adequate conditions (Filgueiras and Thelma Maria, 2003).

Inspecting wind plant constructions is fundamental for the effectiveness and control of public policies. In Brazil, ANEEL is responsible for regulating the expansion of installed capacity and monitoring the progress of plant construction (Orlandi et al., 2021). However, inspection is carried out directly on-site through the displacement of qualified professionals at high costs. The expectation of wind energy growth tends to have many projects with low energy production, increasing the number of processes to be evaluated and urgently requiring process automation.

Periodic satellite images are a promising tool for monitoring works in the electricity sector, being a low-cost alternative for the surveillance of construction stages. Therefore, remote sens-

ing studies for the detection of infrastructure in the electricity sector have recently increased, especially in solar energy mapping: urban photovoltaic solar panels (Bradbury et al., 2016; Jie et al., 2020; Zhuang et al., 2020), water photovoltaic (Xia et al., 2022), and photovoltaic solar plants (Costa et al., 2021; Plakman et al., 2022; Zhang et al., 2021). On the other hand, wind farm detection works are scarce because they are narrow objects and complex arrangement, need for high resolution imaging and use of high-performance detection methods. Considering a continental country, automatic detection is crucial to minimize human activity, such as visual inspection. In this context deep learning methods are the current state-of-the-art for image classification, especially with advances in Convolutional Neural Networks (CNN), which allow the detection of the small, medium, and high-level features (Lecun et al., 2015; Nogueira et al., 2017). In the field of pattern recognition in the electricity sector, methodologies for mapping solar panels have proved to be highly efficient, including by ANEEL itself (Costa et al., 2021).

The main objective of this study is to create a low-cost system using deep learning and remote sensing images to monitor wind farms. The results search to reduce the costs of technical visits and make decisions more quickly and accurately. Since the study areas are extensive, the present research created a personalized pipeline for this task, such as sliding windows and object counting. This methodology is the first to use remote sensing images and artificial intelligence to map wind farms, which can be an avant-garde method to save and enhance public decisions in this sector.

4.2 Materials and Methods

The present research had the following methodological steps (Figure 4.1): (2.1) Data, (2.2) deep learning approach, (2.3) sliding windows using the best model, and (2.4) semantic to instance conversion using GIS.

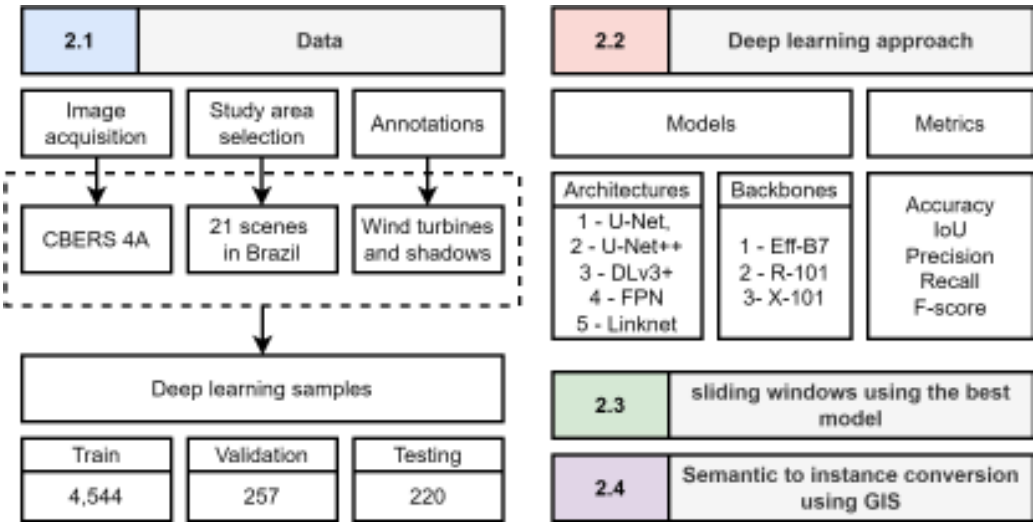

Figure (4.1) Methodological flowchart.

4.2.1 Data

Study Area Selection and Image Acquisition

The study areas seek to represent the main concentrations of wind farms spread across the Brazilian territory (Figure 4.2). In this context, the research covered a wide variety of background landscapes, from coastal areas with the presence of dunes to inland regions with different uses and land cover and covering Brazil from the northeast to the south. This research used the panchromatic images of the China-Brazil Earth Resources Satellite CBERS 4A sensor (2-m resolution), the sixth CBERS family satellite developed by the space technical cooperation between Brazil and China (Vrabel et al., 2021). These images combine the advantages of free distribution (significant cost reduction) and high resolution from the Panchromatic Wide Scan camera. Other possibilities for high-resolution images such as aerial surveys or using orbital satellites (GeoEye-1 (41 cm), WordView-2 (46 cm), WordView-3 (31 cm), WordView-4 (31 cm), Planet Labs (50 cm), QuickBird (61 cm), and IKONOS-2 (1 m)) would represent a significant increase in the cost of monitoring for a country with a continental extension. Besides, other sensors with free data (such as Sentinel-2 or Landsat-8) have difficulties detecting wind farms due to the low resolution. For example, Sentinel-2 images (10 meters resolution) have limitations compared to the CBERS-4A image (Figure 4.3). This study used 21 CBERS 4A scenes throughout the Brazilian territory, incorporating various environments with wind farms in the database. The CBERS 4A provides images with a periodicity of 31 days, bringing monthly updates to each region.

Image Annotation

The mapping of all wind farms for the 21 CBERS 4A scenes used on-screen visual interpretation. The visual interpretation of the wind farm installation considered the following features: (1) foundation concrete, a circular base with a diameter of approximately 20 meters; (2) wind turbines containing blades that rotate the rotor with the force of the wind; and (3) the shadow areas. One of the most significant difficulties in the computer vision community is dealing with small objects, defined as elements with less than 32^2 pixels (Lin et al., 2014; Tong et al., 2020). Although wind farms are a prominent object in height, they are not notable in remote sensing orbital images with a nadir view due to their reduced width. Despite shadows in most remote sensing studies are considered a significant problem as they hide the intended objects (de Carvalho et al., 2021a), an unconventional approach integrates the shadow features into the analysis, occupying a more significant area coverage and facilitating the wind farm detection (Shen et al., 2017).

Deep learning samples

Orbital remote sensing images have extensive dimensions, requiring in the classification and training the subdivision of the images into small patches with a dimension of 128x128 pixels, suitable for including a wind farm. Every wind plant had at least one 128x128 sample, and we

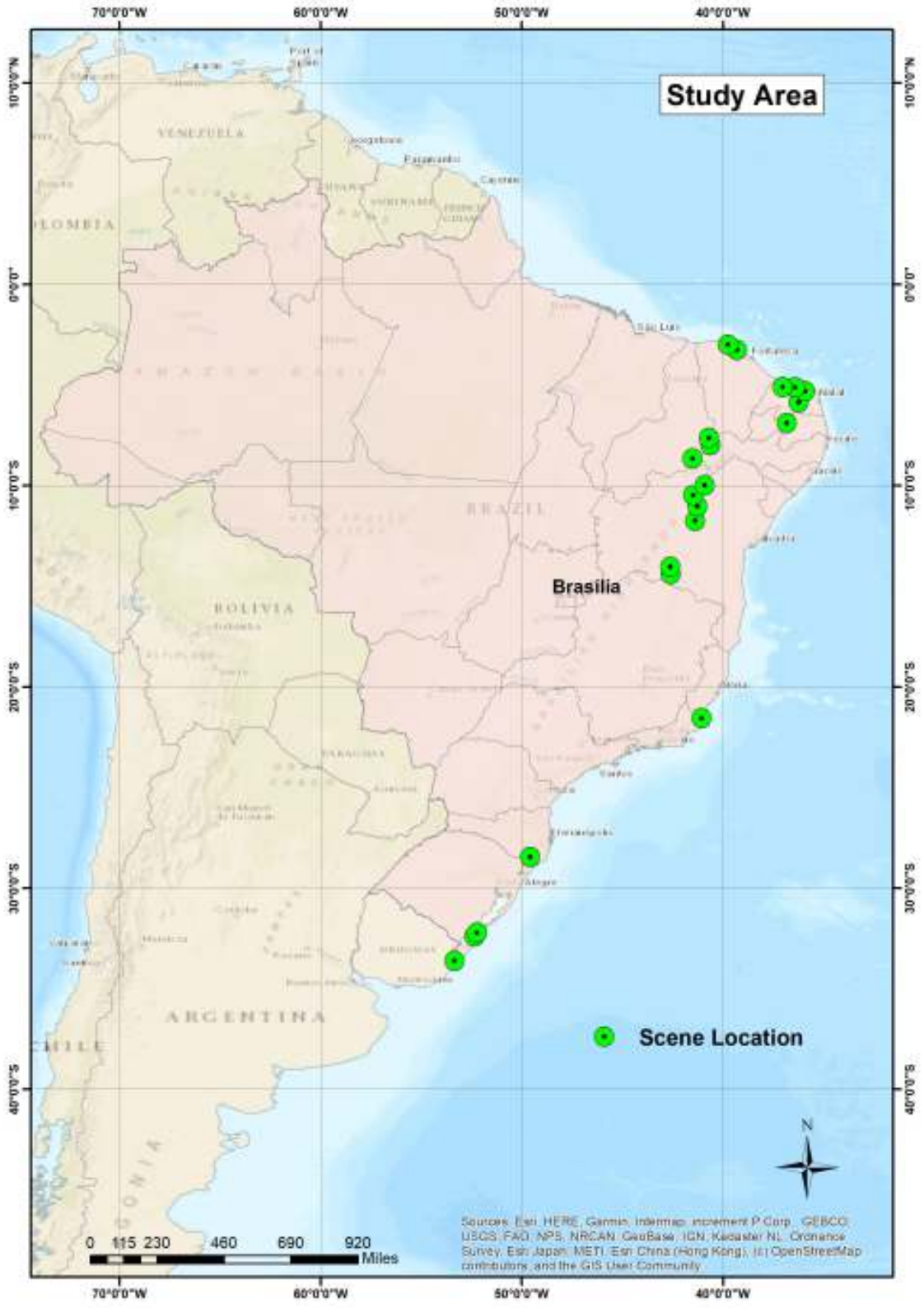

Figure (4.2) Study area.

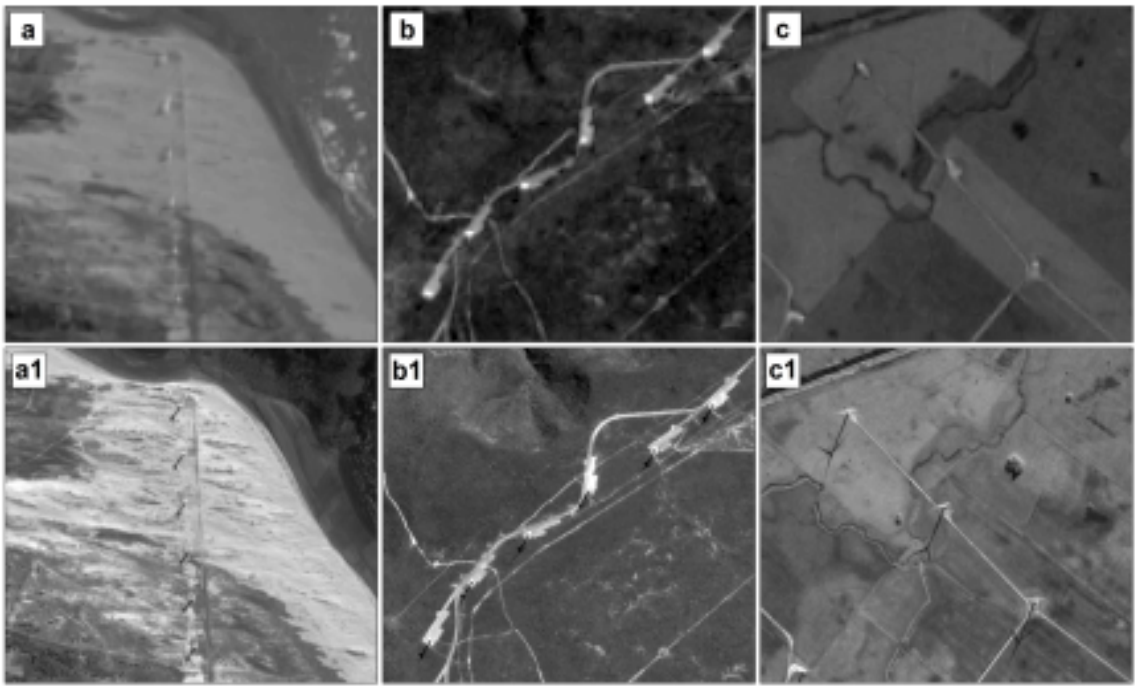

Figure (4.3) Example of the shadows produced by the wind plants considering Sentinel (a, b, and c) and CBERS 4A images (a1, b1, c1).

added at least 20 samples using background-only information. From the 21 CBERS-4A scenes, 14 were for training, 3 for validation, 3 for testing, and an additional scene for evaluation of the sliding window procedure (Table 4.1). The final dataset included 4544, 257, and 220 patches for training, validation, and testing, respectively.

4.2.2 Deep learning approach

Deep learning models

Instance segmentation models such as the Mask-RCNN (He et al., 2020) are the primary approach for recognizing individual objects at a pixel level. However, instance segmentation models for orbital remote sensing may present additional difficulties regarding semantic segmentation: (1) more structured information data requirement (e.g., COCO (Lin et al., 2014)); (2) increasing object detection parameters (e.g., anchor boxes) and procedures (e.g., ROI alignment); (3) image reconstruction by sliding windows becomes challenging; and (4) worse pixel metrics, especially for small objects. Since the wind farms and their shadows do not touch each other, it is simple to convert semantic features to instance features using post-segmentation methods (de Carvalho et al., 2021a; Mou and Zhu, 2018).

Therefore, we used semantic segmentation models to classify all input image pixels (Garcia-Garcia et al., 2017; Guo et al., 2016). The models usually present a structure with a contraction (extraction of meaningful features) and extension (restoring the image dimension) paths. Since this is the first deep learning approach for this target, this investigation compared five state-of-the-art semantic segmentation architectures: U-Net (Ronneberger et al., 2015a), DeepLabv3+

Table (4.1) Dataset information considering the state, location (latitude and longitude), number of wind plants, number of patches, and the set (train, validation, test or sliding windows (SW)). The dataset considered the following Brazilian states: Bahia (BA), Ceará (CE), Piauí (PI), Rio Grande do Norte (RN), Rio Grande do Sul (RS), and Rio de Janeiro (RJ).

State	Location	# Of wind plants	# Of patches	Train/val/test
BA	42°40'48,852"W 14°4'17,174"S	407	656	Train
BA	41°27'37,008"W 11°51'22,643"S	113	290	Train
BA	41°15'58,126"W 11°2'6,711"S	250	228	Train
BA	42°35'57,95"W 14°24'38,101"S	303	377	Train
BA	41°23'53,288"W 10°31'20,718"S	225	251	Train
BA	40°42'49,748"W 7°40'6,644"S	270	288	Train
CE	39°42'39,391"W 3°4'52,63"S	174	250	Train
CE	39°19'57,904"W 3°16'45,851"S	233	315	Train
PI	41°32'16,008"W 8°39'0,225"S	309	323	Train
RJ	41°4'37,302"W 21°34'28,83"S	18	45	Train
RN	36°26'53,947"W 5°14'40,991"S	203	285	Train
RN	35°55'58,156"W 5°20'52,179"S	818	836	Train
RS	53°19'10,675"W 33°35'48,462"S	305	340	Train
RS	49°35'52,656"W 28°27'51,075"S	60	60	Train
BA	40°58'22,055"W 10°5'0,823"S	113	124	Validation
PB	36°43'49,184"W 6°58'1,276"S	59	101	Validation
RS	52°13'4,441"W 32°13'40,063"S	32	32	Validation
PI	40°37'23,185"W 7°59'46,973"S	98	118	Test
RN	36°12'37,606"W 5°44'52,836"S	53	62	Test
RS	52°21'33,646"W 32°25'18,929"S	40	40	Test
RN	37°2'11,031"W 5°7'59,359"S	382	-	SW test

(Chen et al., 2018), Feature Pyramid Network (FPN) (Lin et al., 2017), and U-Net++ (Zhou et al., 2018b). Furthermore, the model evaluation considered three backbones: Efficient-net-B7 (Tan and Le, 2019a), ResNeXt-101 (Xie et al., 2017), and ResNet-101 (He et al., 2016). The hyperparameters remained the same to ensure cohesion between the different models: learning rate of 0.0001, batch size of 20, and 100 epochs. The image processing used a computer equipped with an NVIDIA RTX 3090 and i9 processor for all experiments.

Accuracy metrics

Most of the accuracy metrics of semantic segmentation models come from the confusion matrix. As our problem is binary, there are four possibilities: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). However, wind farms and their shadows, being small objects, have a high percentage of background data, increasing the presence of true negatives. Thus, the main metric for our analysis is the Intersection over Union (IoU), expressed by TP((TP+FN+FP)). This metric considers both types of errors (FP and FN) and does not consider the TN. Even though this metric is the most relevant, we also evaluated the overall accuracy ((TP+TN)((TP+TN+FN+FP))), precision (TP((TP+FP))), recall (TP((TP+FN))), and f-score ((2*precision*recall)((precision+recall))).

4.2.3 Sliding windows for large image classification

Large-scale image segmentation should use a sliding window (SW) approach, considering an image subdivision into sizes equivalent to the training samples with an overlap area demarcated by the stride value between the windows. Thus, the SW approach performs the classification in sequential frames (from left to right and top to bottom). The stride value in semantic segmentation is essential, where smaller stride values bring greater overlap area and better results as the average of overlapping pixels minimize errors (da Costa et al., 2021b; de Albuquerque et al., 2020b). However, the computational cost also increases. The situation in question presents a trade-off in performance and computational cost.

Most SW studies consider amorphous and large targets, and few studies have compared the different strides for small objects. This investigation evaluated four stride values (16, 32, 64, and 128), considering an independent scene with a high concentration of wind power stations. The predictions generate probabilities between 0 and 1, where the average overlapping pixels improve the accuracy. For this purpose, the ranking metrics that evaluate classifiers over variable thresholds are adequate, so we used Precision Area-Recall Under the Curve (PR-AUC Receiver Operating Area Under the Curve (ROC AUC).

4.2.4 Semantic to instance segmentation conversion using GIS

In the present study, the conversion from semantic to instance features allows fast object counting using an approach from two studies developed for car detection (de Carvalho et al., 2021a; Mou and Zhu, 2018). The semantic segmentation of contacting targets can be undesirably grouped into a single element. The cited studies isolate the grouped objects by inserting a buffer around

the individual objects, generating a result that separates the inside of the objects containing individual attributes. In our study, an attractive property of wind farms is that they are always far enough away from each other. In other words, there will never be a scenario of merging many predictions into a single polygon. For this reason, we do not need to add borders to objects to be able to separate them.

This process can be easily adapted to GIS platforms like ArcGIS by applying a raster procedure to the polygon. As the elements are far from each other, the number of polygons tends to be the number of wind farms. However, there may be noisy predictions in some semantic segmentation results, misleading the counting procedure. Thus, we can eliminate polygons represented by a certain number of pixels. In this study, we eliminated polygons with areas below $350m^2$, since the wind plants present on average more than $800m^2$.

4.3 Results

4.3.1 Model evaluation and comparison

Table 4.2 lists the results considering the different architectures and backbones. The IoU and F-score are usually the most appropriate in choosing the best model since they consider FP and FN errors. For both IoU and F-score, the best model used the LinkNet architecture with the Eff-B7 backbone, but the U-Net and U-Net++ models presented similar scores. DLv3+ and FPN presented more than 3% difference in the best models from the other three. Interestingly, only three of the 15 different models presented a recall score higher than the precision score. The accuracy analysis shows to be very misleading since most of the pixels are background, and most models presented very high scores near 100%.

Figure 4.4 shows examples from the test set for the best model (LinkNet with the Eff-B7 backbone). The results clearly demonstrate that the models could understand distinct shadow representations, which is very accurate for mapping wind plants. Nonetheless, there are some spots in which the algorithm may bring some errors. Figure 4.5 shows three examples of possible errors that may occur. The first row shows lookalike features, erroneously detecting a wind plant shadow. The second and third examples show discontinuity errors with relevance in the raster to polygon conversion due to the possibility of giving misleading results.

Table 4.3 lists the training period for each model and the inference time on a single 128x128 frame. For DeepLabv3+, U-Net, FPN, and LinkNet, the training period for a single epoch presented a similar behavior among the three backbones, in which Eff-B7 > X-101 > R-101. The U-Net++ had a higher training period for X-101 than the rest. Note that the overall behavior tends to be preserved, but changing the computer configurations may vary the results.

Table (4.2) Dataset information considering the state, location (latitude and longitude), number of wind plants, number of patches, and the set (train, validation, test or sliding windows (SW)). The dataset considered the following Brazilian states: Bahia (BA), Ceará (CE), Piauí (PI), Rio Grande do Norte (RN), Rio Grande do Sul (RS), and Rio de Janeiro (RJ).

Architecture	Backbone	Accuracy	Precision	Recall	F-score	IoU
DLv3+	Eff-B7	99.58	79.61	79.77	79.69	66.24
	X-101	99.57	78.01	79.96	78.97	65.25
	R-101	99.56	79.01	78.11	78.56	64.69
U-Net	Eff-B7	99.63	83.13	80.31	81.69	69.05
	X-101	99.63	85.17	77.60	81.21	68.36
	R-101	99.61	81.99	78.94	80.44	67.28
LinkNet	Eff-B7	99.66	84.30	82.55	83.41	71.55
	X-101	99.62	82.99	79.04	80.97	68.02
	R-101	99.62	82.39	79.74	81.04	68.13
FPN	Eff-B7	99.59	80.28	79.20	79.73	66.30
	X-101	99.58	79.11	79.39	79.25	65.63
	R-101	99.58	80.41	78.46	79.42	65.87
U-Net++	Eff-B7	99.64	83.86	80.74	82.27	69.88
	X-101	99.63	85.17	77.60	81.21	68.36
	R-101	99.61	81.99	78.94	80.44	67.28

Table (4.3) Training period (TP) (in seconds), and inference time (IT) (in milliseconds) considering a computer equipped with an NVIDIA RTX 3090 (24 GB RAM) with an i9 processor.

Architecture	Backbone	TP (s)	IT (ms)
DLv3+	Eff-B7	65	42.98
	X-101	40	21.58
	R-101	23	14.50
U-Net	Eff-B7	58	48.17
	X-101	40	23.16
	R-101	28	16.44
LinkNet	Eff-B7	62	44.44
	X-101	38	22.82
	R-101	29	16.98
FPN	Eff-B7	60	44.77
	X-101	37	23.27
	R-101	27	16.81
U-Net++	Eff-B7	64	43.30
	X-101	65	21.81
	R-101	48	16.62

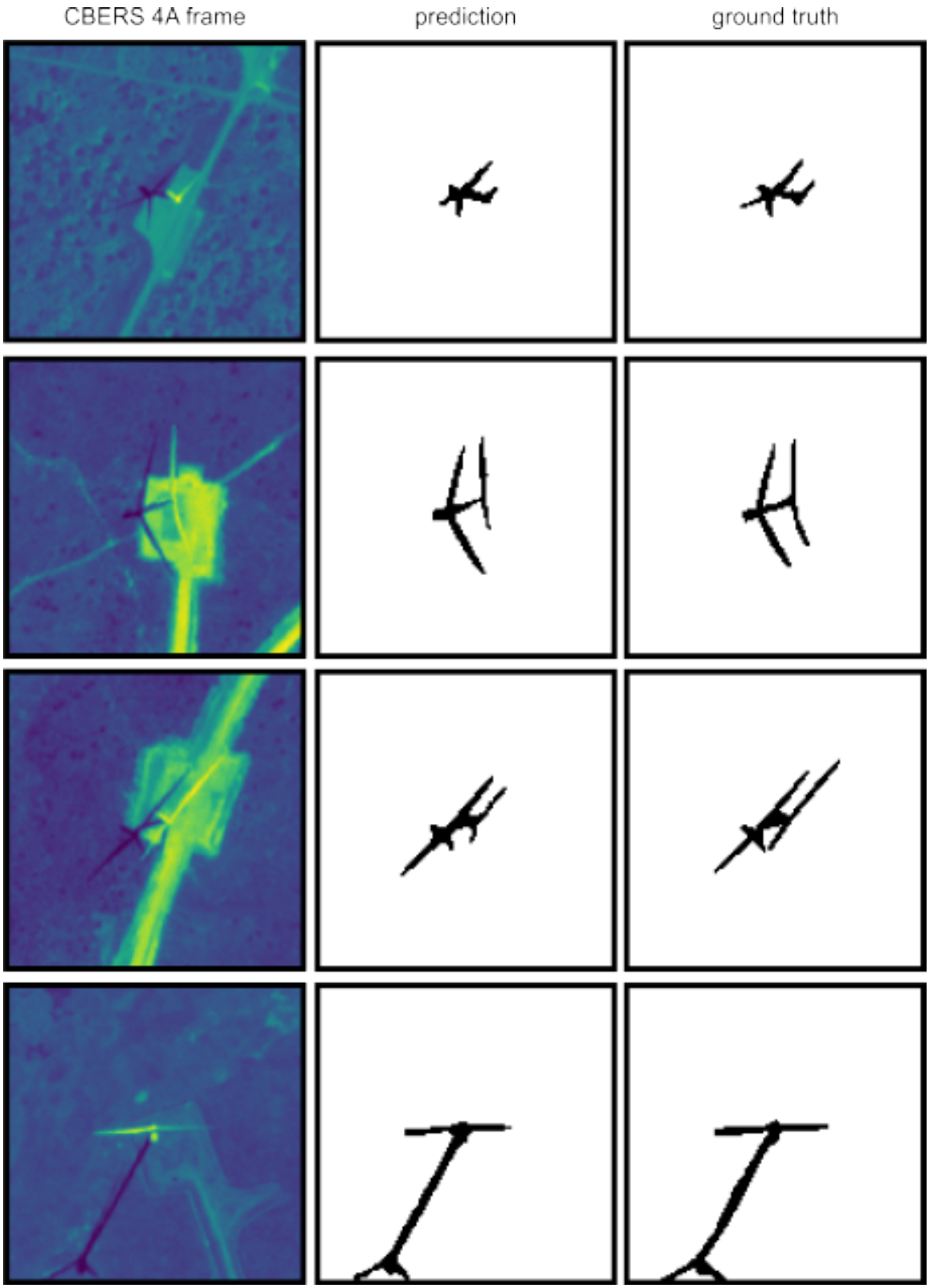

Figure (4.4) Image patches from the test set considering the original CBERS 4A image, the deep learning prediction, and the ground truth image.

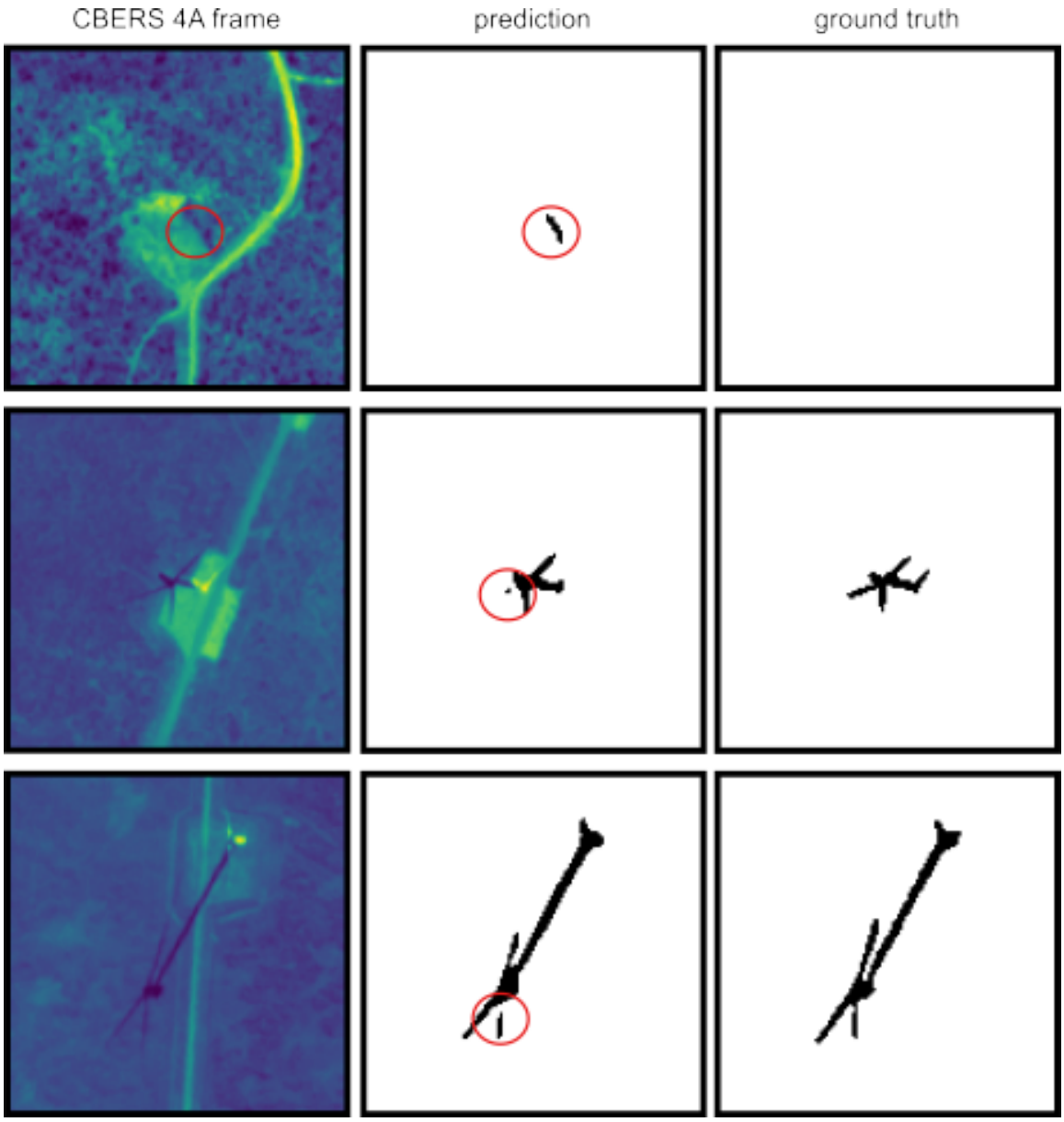

Figure (4.5) Image patches from the test set considering the original CBERS 4A image, the deep learning prediction, and the ground truth image. The spots in red are highlighted areas that show in more detail the areas with errors.

4.3.2 Sliding windows approach

Table 4.4 lists the results considering different stride values for the ROC AUC and PR-AUC scores. The scene presented 19,968x19,968-pixel dimensions, and varying the dimensions would directly affect the mapping time since the number of necessary iterations would change. This scene using a 128-pixel stride, which corresponds to no overlapping pixels, takes nearly 22 minutes to complete. The time necessary quickly escalates when reducing the stride. The time nearly quadruplicated when reducing the stride by two. Within those tests, the metrics keep climbing when reducing the strides. However, the improvement tends to get lower each time. Figure 4.6 shows some differences in predictions with distinct strides. The quality of the data

segmentation improves by decreasing the stride, but the main information is knowing where the wind farms are. Thus, the stride choice for practical applications will depend on the types of errors present (such as continuity errors) and the computational resources.

Table (4.4) Receiver Operation Characteristic (ROC AUC), Precision-recall (PR AUC) area under the curve, Intersection over Union (IoU), and mapping time using different stride values.

	Stride 16	Stride 32	Stride 64	Stride 128
ROC AUC	98.23	97.96	95.94	94.03
PR AUC	87.22	85.41	82.27	71.68
IoU	69.38	68.95	66.28	60.78
Mapping time (hr:min:sec)	22:01:21	5:30:20	01:22:32	00:21:58

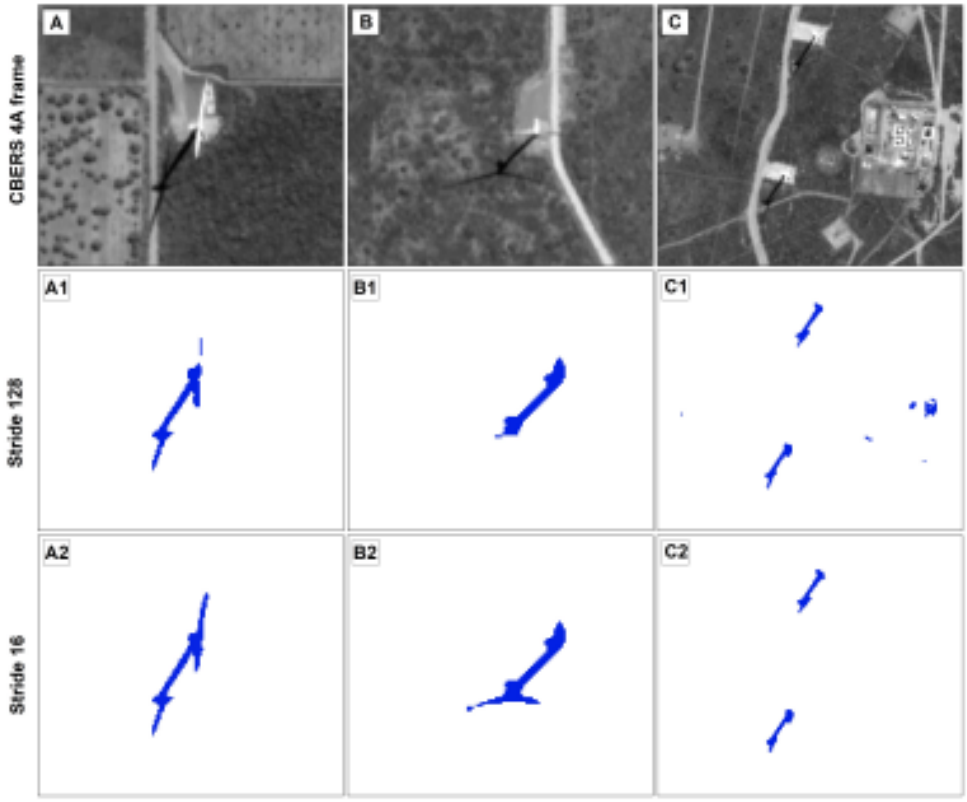

Figure (4.6) Differences in the results from the sliding windows approach using different stride values.

4.3.3 Final GIS classification

Figure 4.7 shows the final representation with the targets in shapefile format after the raster to polygon operation. Noisy representations are prevalent errors that, in this situation, would bring misleading results since we can estimate the number of wind power plants as the number of polygons. Noisy polygons are predominantly much smaller than those in wind farms. Wind farms average more than 900m2, while errors are generally less than $350m^2$. Thus, elimination using a size threshold value is a viable solution to avoid this type of error.

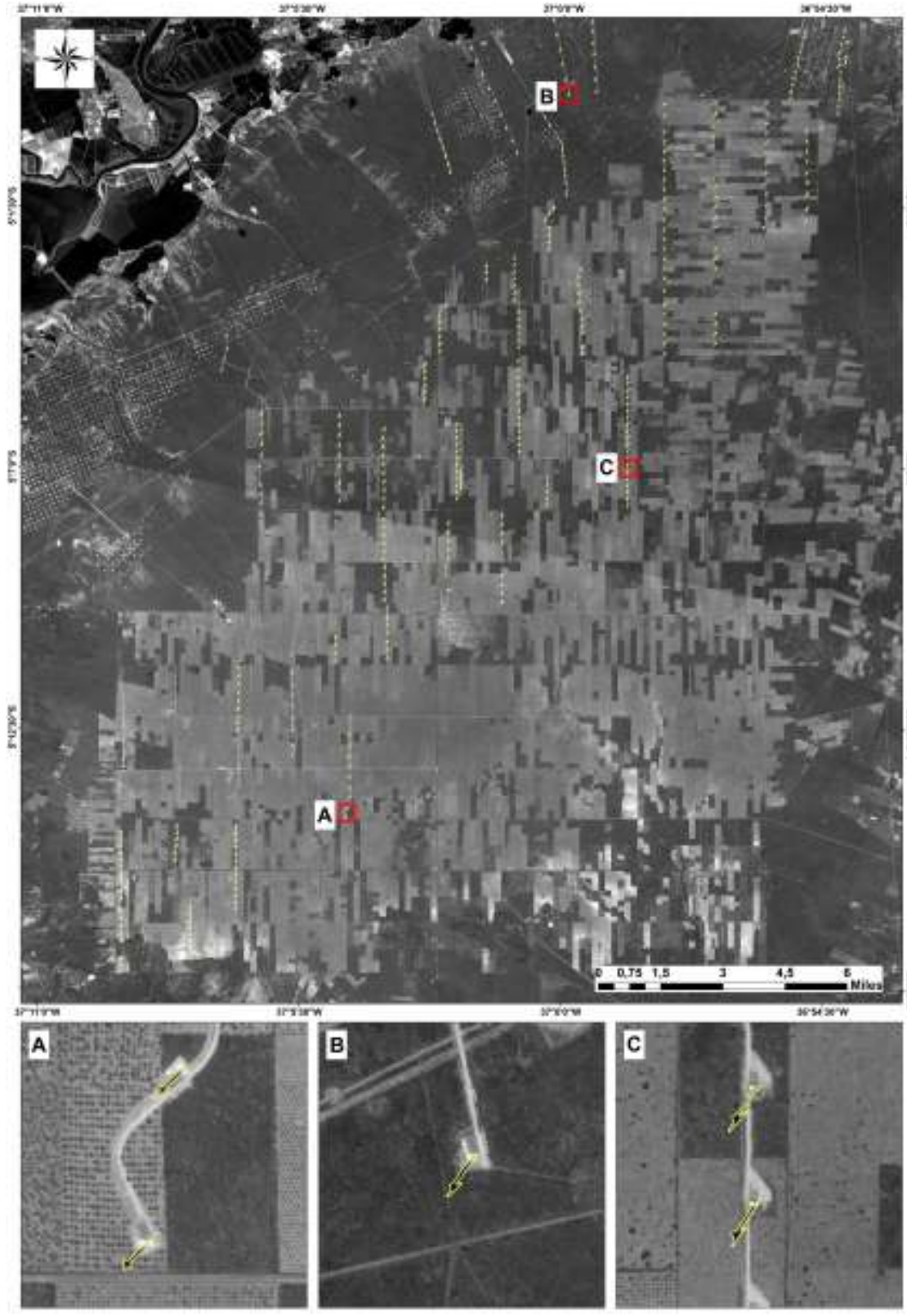

Figure (4.7) Results using GIS software, in which the different colored objects represent different instances from wind plants.

To prove this case, Table 5 lists the per-object results, in which the accuracy is over 90%, showing the efficiency of the overall method. The elimination procedure significantly impacts this kind of analysis since the total number of eliminated noisy features was 1,092, which would lower the presented metrics considerably and provide misleading results for inspection and decision-making. Another interesting result is the absence of false negatives. False positives are shadow-like features of wind farms that are difficult even for humans to identify. Figure 4.8 shows four examples of false positive errors, in which 4.8A is a similar tower structure, and 4.8B, 4.8C, and 4.8D are preliminary constructions in the location of wind plants, which presents similar structures and shadows.

Table (4.5) Results for the final prediction.

Metric	Result
True Positives	369
False Positives	37
False Negatives	0
accuracy	90.88

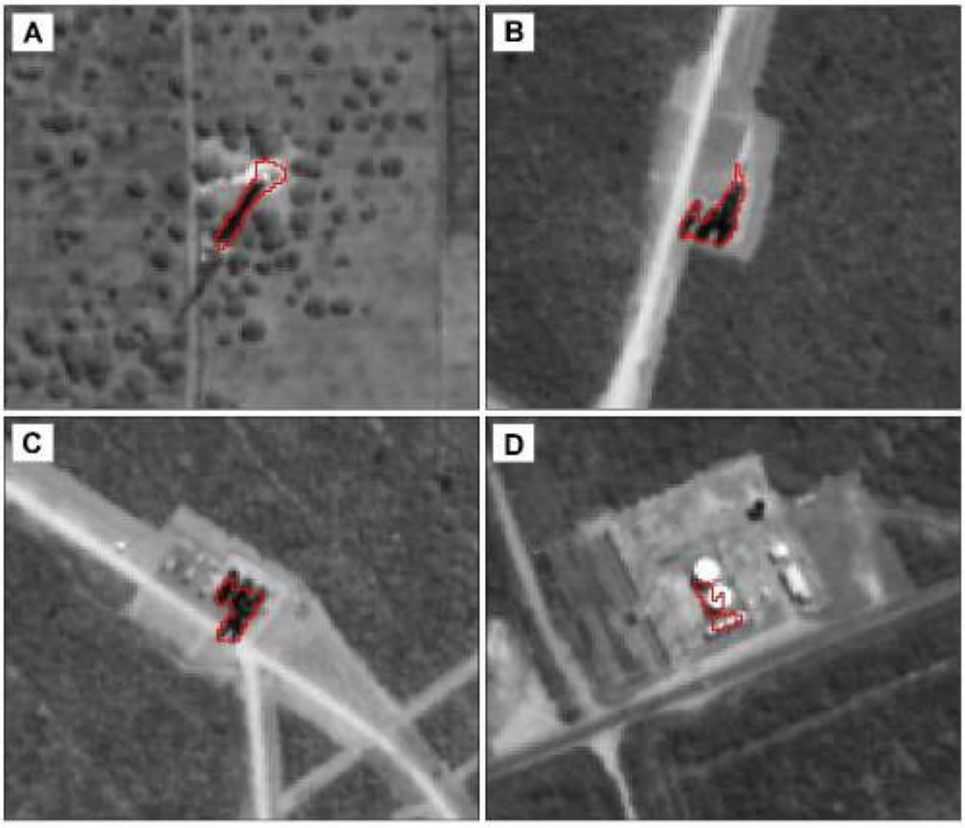

Figure (4.8) Four examples of overestimation errors in the final classification.

4.4 Discussion

The diversification of the Brazilian energy mix with renewable sources and the establishment of alternatives to hydroelectric plants are fundamental strategies to be in line with the commitment to reduce carbon emissions of the 2030 Agenda and minimize the dependence on hydropower, which could bring energy security concerns in case of long periods of drought. In this sense, the expansion of energy production needed to meet future demands will rely on more decentralized and intermittent sources such as wind and solar. The construction of wind farms has increased significantly due to the vast resource of this natural source in the Brazilian territory and government actions to reduce the risk that allows a high power generation capacity at competitive costs (Rego and de Oliveira Ribeiro, 2018; Simas and Pacca, 2014). The recent prospect of accelerated growth in wind generation capacity makes it imperative that regulatory agencies invest in technological innovations that quickly satisfy regulatory demands. In order to ensure the production capacity, it is essential that the Government and market players can track and oversee the progress of the construction of new wind power sites.

Developing a technological system based on remote sensing images and deep learning establishes an important tool and database capable of continuous monitoring that can improve inspection with reduced human work and promote other innovations in spatial analysis. Besides, ongoing surveillance based on free remote sensing data encourages investors to adhere and comply with the regulatory process and allows special attention to be given to disclosing information on investments in wind energy production. Spatial information, constantly updated and available for public consultation, allows the temporal evaluation of investments in infrastructure, favoring investors, community, and public agencies in planning significant decisions in the electricity sector.

Therefore, this research establishes an automated pipeline solution for monitoring the construction of wind farms using remote sensing and deep learning methods, achieving low costs, high frequency, and coverage of large areas. Furthermore, this is the first remote sensing study that uses deep learning architectures to detect wind farms by establishing an extensive database containing a wide distribution in tropical scenarios. This first deep learning dataset for wind energy is publicly available and could be an effective tool for application in other regions of the world. The results of the different CNN models reach a high accuracy, where the best model was the LinkNet architecture with Eff-B7 backbone. However, the U-Net and U-Net++ models also using the Eff-B7 backbone obtained close accuracy metrics. Unlike other targets such as cars, buildings, or solar plants, the detection of wind farms with their inherent shadows relativizes the error found, as it is not the real representation of the intended object. Therefore, the pixel-level error representing the fine adjustments to the edges of objects is less relevant than the object-level error that verifies the presence or not of the wind farm. This error analysis is also suitable for image reconstruction using the sliding window, where results with a stride value of 64 may be adequate considering the processing time. The image reconstruction results are compatibles with other studies (da Costa et al., 2021b; de Albuquerque et al., 2020b), improving accuracy with decreasing stride value.

Finally, the proposed pipeline generates results categorized as instance segmentation from the coupling of semantic segmentation models with GIS applications. The nature of the target (sufficiently separate) allows for a simple, inexpensive, and quick approach to segmenting wind farm instances. This approach differs from other targets, such as vehicle detection, which requires additional steps to recognize individual objects (de Carvalho et al., 2021a; Mou and Zhu, 2018).

4.5 Conclusion

The present research aimed to create an entire pipeline using deep learning, remote sensing, and GIS platforms to detect wind plants. The deep learning section provided an extensive database with a wide intraregional characteristic in the full extension of the Brazilian territory. This database may be used to train new models and apply them in different countries. Even if the characteristics in other countries seem too different, it can be used as a transfer learning mechanism, requiring fewer samples from other countries to work well. We compared five architectures and three backbones, totaling 15 different models, in which we found that the LinkNet architecture with the Efficient-net-B7 backbone provided the best results. We have configured this instance segmentation problem as a semantic segmentation problem since the target characteristics allow easy separation, and semantic segmentation models are simpler and easier to compare. The sliding windows approach showed an essential tradeoff between performance and computational time, which can be adjusted depending on the demands. The GIS is the last part that brings the essential and relevant information by simply using a raster to polygon procedure and eliminating little polygons, usually noisy features. This entire pipeline is straightforward and can be easily adapted to other targets in the electric sector, such as power plants, and in other countries and regions.

Chapter 5

Box-Free Instance Segmentation with Touching Objects

This chapter introduces an alternative procedure to obtain instance-level predictions without the complexity of traditional instance segmentation models, which require bounding boxes, classification, and segmentation. Our alternative procedure uses semantic segmentation models with borders in the data preparation. This chapter addresses this topic on extremely crowded objects, in our case, vehicles. The results from this chapter were submitted to IEEE Journal of Selected Topics in Applied Remote Sensing, and it is currently in the second round of review.

5.1 Presentation

Usually, the city's infrastructure was not designed to absorb population growth and road traffic, which has reached high congestion levels in many urban centers worldwide. The accentuated growth in the number of vehicles makes monitoring and managing urban traffic highly complex and necessary. In this context, automatic vehicle detection based on remote sensing images is a powerful tool for various applications such as traffic monitoring, air pollution, congestion studies, public safety, parking utilization, disaster management, and rescue missions. Periodic image acquisition provides information on the number and location of vehicles in different urban environments, allowing coverage of large areas and proper monitoring of moving targets.

Vehicle detection is a widely studied topic in the computer vision community, containing several studies with ground-view and aerial-view images. These two approaches present marked differences in vehicle representation, in which ground images emphasize the vehicle faces, while the top view of the vehicle acquires straight shapes (Ji et al., 2019a; Sakhare et al., 2020). Another significant difference is that the vehicle's spatial resolution in aerial images is significantly lower than in terrestrial images. In-ground view images, several literature reviews address advanced driver assistance systems (ADAS) for autonomous vehicles using image processing and vehicle detection from various onboard handling sensors such as radar, monocular camera, and camera binocular (Feng et al., 2021; Janai et al., 2020; Wang et al., 2019). In addition, several

studies use images from surveillance cameras on roads (Song et al., 2019), on top of buildings (Xi et al., 2019), pedestrian bridges (Fachrie, 2020), among others.

Despite the broad applicability of ground images and videos, vehicle detection from high-resolution aerial and satellite images allows for a synoptic understanding of city patterns, guiding crucial public policies such as urban planning and traffic management. Vehicle detection using aerial view imagery includes different strategies and sensors such as unmanned aerial vehicles (UAV), airplanes, or orbital platforms, which provide data at different heights and resolutions.

Even though skilled professionals may easily distinguish vehicles from different urban features, the rapid and automatic classification is a challenging task since the vehicles: (a) are small objects; (2) present high variability in shape, color, and size; (3) appear in different background settings; (4) present different brightness and contrasts among the city; (5) may be crowded (e.g., parking lots); (6) may be occluded by other objects, such as trees and buildings; and (7) have many look-alikes in the city. Figure 5.1 shows six examples of difficult areas to identify the vehicles, where A and B present shadows, C and D show a large concentration of vehicles, and E presents look-alikes (the tombs are very similar to cars when seen from this angle), and F presents occluded cars by the building roof.

Figure (5.1) Six examples (A, B, C, D, E, and F) of difficult regions to classify cars in the urban setting.

The Deep Learning (DL) methods currently represent state-of-the-art vehicle detection, surpassing traditional algorithms. These advances are strongly related to Convolutional Neural Networks (CNN), which apply kernels along with the image, obtaining low, middle, and high-level features and enhancing the classification results. Vehicle detection using deep learning may

present different approaches, such as object detection (Zhao et al., 2019), semantic segmentation (Guo et al., 2018), and instance segmentation Hafiz and Bhat (2020). In object detection, the DL outputs bounding boxes around the car. Instance segmentation generates bounding boxes and a segmentation mask, and semantic segmentation outputs a class-aware segmentation mask.

Most studies on vehicles address object detection that focuses on delineating the targets' bounding box, while instance segmentation, which aims at mapping each object at the pixel level, is still little explored. A challenge in the individual segmentation of vehicles is the lower performance for small objects that, when they are very close, coalesce into a single group (Mou and Zhu, 2018; Tayara et al., 2018). Furthermore, deep instance segmentation methods require a large amount of data, especially considering small object detection. Therefore, training requires a much more complex annotation (since it requires the polygons from each object), containing all possible variations and apparition locations to not depend on a given scenario. The Common Objects in Context (COCO) (Lin et al., 2014) dataset defined small objects with less than 32^2 pixels and results considering the small objects are nearly half of the performance of medium and large objects.

More recently, artificial intelligence has an upcoming trend that aims to enhance results and practical solutions by using a data-centric rather than a model-centric approach. The central concept behind this is that the model performance is already very high and that enhancing the data would bring better benefits. One pillar of the model-centric approach is the selection of more informative samples within the dataset. In this context, active learning is a promising methodology to obtain quality labeled data sequentially. In remote sensing, images often present vast dimensions, and the integration of commonly used GIS software may be an excellent ally for active learning in object detection since: (a) we may see the entire data at once, (b) It is very straightforward to manipulate and correct polygon data, and (c) we may use other facilities such as polygon shapefiles to choose where to gather the data.

The present research aimed to advance in three fields (data generation through iterative learning, deep learning method, and dataset):

- **Iterative learning procedure for data generation**: A novel proposition for integrating DL with commonly used GIS software by iteratively correcting erroneous areas, being less time-consuming and laborious.
- **Bounding Box-Free instance segmentation**: a novel instance segmentation method that uses object interiors and contours to isolate them and output separate instances.
- **BSB Vehicle Dataset**: A city-scale dataset with polygons shapefiles.

5.2 Related Works

In the last two decades, different strategies have been developed and described for vehicle detection through aerial and orbital images. In this trajectory, two main approaches stand out (Li et al., 2020a; Shen et al., 2021; Shi et al., 2021): (a) methods based on superficial learning and (b) deep learning-based methods.

5.2.1 Early vehicle detection studies using a shallow-learning based approach

Considering vehicle detection approaches based on superficial learning, Hinz (2003) proposed a generic subdivision into explicit and implicit models. The explicit model describes a vehicle in 2D or 3D (representation of a box or wire-frame structure), considering the car detection from a "top-down" or "bottom-up" model. The implicit model considers the collection of multiple features of a region of the image and their statistics gathered in vectors followed by a classification process (single classifier, combination of classifiers, or hierarchical model). In the present analysis, we considered the following groups of algorithms: (a) pixel-wise classification and segmentation (including threshold segmentation method, segmentation based on pixel clustering, segmentation based on edge detection and region growth method, segmentation based on inter-frame difference or background difference); (b) object-based classification; object detection (obtaining the bounding box without vehicle segmentation) from multiple features and machine learning within a sliding window approach.

The threshold segmentation method was widely used in different pre-processed images to highlight vehicles, such as Principal Component Analysis (PCA), Bayesian Background Transformation (BBT), and gradient-based method (Sharma et al., 2006); Morphological grayscale method and background difference (vehicle enhancement by subtraction between the original image and the road background image) (Zheng et al., 2013). Cheng et al. (2012) perform pixel-wise classification for vehicle detection using Dynamic Bayesian Networks (DBNs), considering features that comprise pixel-level information and the relationship between neighboring pixels in a region (location analysis of features and color attributes).

Object-based methods use image segmentation to split an image into separated regions and classify them instead of pixels (Hossain and Chen, 2019). Different vehicle detection surveys use object-oriented image classification, considering: (a) eCognitionő classification (Holt et al., 2009); (b) segmentation using Otsu Threshold, feature extraction (geometric-shape properties, gray level features, and Hu moments), and statistical classifier (Eikvil et al., 2009); and (c) superpixel-based image segmentation, HOG features, and SVM (Chen et al., 2016).

Vehicle detection methods have increased significantly by combining more robust descriptor extraction procedures with machine learning methods for object detection (Table 5.1). Therefore, vehicle detection uses an image scan through a pre-trained classifier. Among the methods of extraction and selection of features, the most used were: Haar-like features, Histogram of Oriented Gradient (HoG), Histogram of Gabor Coefficient (HGC), and Local Binary Patterns (LBP), Local Steering Kernel (LSK), bag-of-words (BoW) and Scale Invariant Feature Transform (SIFT). Several studies have improved the description of cars by combining different re-

source extraction methods. The most used machine learning methods were the Support Vector Machines (SVM) and Adaptive Boosting (AdaBoost) in the classification step. However, the literature also describes the use of other methods to compare and improve detection accuracy and efficiency, such as k-Nearest Neighbor (k-NN), Decision Trees (DT), Random Forests (RF), Dynamic Bayesian Network (DBN), Partial Least Squares (PLS). Some associations between feature extraction methods and classifiers had more significant propagation for detecting vehicles such as HoG + SVM (Dalal and Triggs, 2005) and Haar-like + AdaBoost called Viola-Jones (Viola and Jones, 2001). However, the shallow-learning-based methods do not sufficiently describe and generalize vehicle detection in complex backgrounds. Some studies to minimize errors have restricted vehicle detection to certain circumstances: (a) only along roads, considering the use of masks from a buffer area (Leitloff et al., 2014; Moranduzzo and Melgani, 2014a,b; Nguyen et al., 2007; Zheng et al., 2013); (b) exclusion of objects elevated above a certain height from the DEM (e.g., buildings and vegetation) (Tuermer et al., 2013); and correlation of cars in consecutive frames (Cao et al., 2011). Also, most of these methods are sensitive to the in-plane rotation of objects (detecting only in a specific orientation) and to changes in lighting, such as Viola-Jones.

In the transition from traditional to DL methods, some studies use deep architecture only to extract highly descriptive features combined with a machine learning classifier. In this approach, the following propositions stand out: Deep Boltzmann Machines (DBMs) and weakly supervised learning (Han et al., 2015), multilayer deep resource generation model using DBMs and Multiscale Hough Forest Model (Yu et al., 2015, 2016), CNN and Exemplar-SVMs (Cao et al., 2016), and CNN and SVM (Ammour et al., 2017).

5.2.2 Deep learning-based vehicle detection

A significant milestone in CNN's dominance in computer vision was its success in the ImageNet Large Scale Visual Recognition Challenge in 2012 (Krizhevsky et al., 2017). DL-based vehicle detection studies have intensified in the following years, with an annual increase making it the dominant method today. Deep learning architecture networks perform better than shallow learning-based methods due to the following reasons (Sevo and Avramovic, 2016): (a) operates both for feature extraction and classification; (b) CNN improves automatic feature generation with the ability to learn local characteristics of different orders, inherently exploiting spatial dependence; and (c) less time-consuming. Different deep learning approaches have been applied in vehicle detection, such as object detection, semantic segmentation, and instance segmentation.

Object Detection

Vehicle studies using object detection are dominant due to fast target detection, improving real-time monitoring efficiency. However, these methods do not allow a precise mapping of their contours obtained with semantic and instance segmentation. Table 5.2 presents the main studies of vehicles using object detection methods. A subdivision of the object detection algorithms is two-stage object detection and one-stage object detection.

Table (5.1) Studies developed for the detection of cars using different feature extraction approaches (shallow-learning-based features) and classification, in which the feature extraction methods described are: Color Probability Maps (CPM), Haar-like features (Hlf), Histogram of Gabor Coefficients (HGC), Histogram of Oriented Gradients (HoG), Local Binary Patterns (LBP), Local Steering Kernel (LSK), Local Ternary Pattern (LTP), Opponent Histogram (OH), Scale Invariant Feature Transform (SIFT), Integral Channel features (ICFs), Bag-of-Words (BoW), Vector of Locally Aggregated Descriptors (VLAD), Pairs of Pixels Comparisons (PPC), Road Orientation Adjustment (ROA), Template Matching (TM), and Hough Forest (HF). The classification methods are Adaptive Boosting (AdaBoost), Decision Trees (DT), Deformable Part Model (DPM), Dynamic Bayesian network (DBN), k-Nearest Neighbor (k-NN), Partial Least Squares (PLS), Random Forests (RF), and Support Vector Machines (SVM). The images used in the articles are Unmanned Aerial Vehicle (UAV), Google Earth (GE), and Wide Area Motion Imagery (WAMI).

Article	Features	Classifier	Image
Leberl et al. (2007)	HoG, Hlf, LBP	AdaBoost	aerial
Nguyen et al. (2007)	HoG, Hlf, LBP	AdaBoost	aerial
Grabner et al. (2008)	LBP, HoG, and Hlf	AdaBoost	aerial
Cao et al. (2011)	HoG	SVM	UAV
Gleason et al. (2011)	HoG and HGH	k-NN, SVM, DT, and RF	aerial
Kembhavi et al. (2011)	HoG, CPM, PPC	PLS	aerial
Liang et al. (2012)	HoG and Hlf	AdaBoost and SVM	WAMI
Shao et al. (2012)	HoG, LBP, and OH	SVM	aerial
Tuermer et al. (2013)	HoG	AdaBoost	aerial
Moranduzzo and Melgani (2014a)	SIFT	SVM	UAV
Moranduzzo and Melgani (2014b)	HoG	SVM	UAV
Leitloff et al. (2014)	Hlf	AdaBoost and SVM	aerial
Liu and Mattyus (2015)	ICFs + HoG	AdaBoost	UAV and GE
Madhogaria et al. (2015)	HoG	SVM and Causal MRF	UAV
Razakarivony and Jurie (2016)	HOG, LBP, and LTP	SVM, DPM, TM, and HF	aerial
Xu et al. (2016)	HoG and Hlf	SVM and AdaBoost	UAV
Cao et al. (2017)	SIFT	Multi-Instance Learning	satellite
Xu et al. (2017)	Hlf + ROA	AdaBoost	UAV
Zhou et al. (2018a)	LSK + BOW	SVM	UAV and satellite
Liu et al. (2019a)	LSK + VLAD	Directed-Acyclic-Graph SVM	aerial

Two-step methods first generate several bounding boxes around potential objects called region proposals, and then a classifier determines the objects presence. The classification of each potential object slows down the process, focusing on detection accuracy. As examples of two-stage object detection algorithms highlight Regions with CNN features (R-CNN) (Girshick et al., 2014), its variants Fast R-CNN (Girshick, 2015), Faster R-CNN (Ren et al., 2017), and Mask R-CNN (He et al., 2020).

One-stage object detection processes images through a single neural network, detecting and classifying multiple objects simultaneously and ensuring speed. These methods focus on the detection speed but have limitations in detecting crowded groups of small objects. Among these algorithms, You Only Look Once (YOLO) (Redmon et al., 2016), You Only Look Twice (YOLT) (Van Etten, 2018), and Single Shot Multibox Detector (SSD) (Liu et al., 2016) are the most prevalent.

Semantic and instance segmentation

Vehicle studies with semantic and instance segmentation present less quantity than those developed with object detection methods. Tayara et al. (2018) performed a Fully Convolutional Regression Network (FCRN), whose training stage uses the input image and ground truth data that describes each vehicle as a 2-D Gaussian function distribution. Therefore, the vehicle's original format acquires a simplified elliptical shape in the ground truth and output images. The vehicle segmentation uses a threshold value in the predicted density map, generating a binary mask. Although the method avoids grouping cars and favors counting, vehicles take on a different form described by the Gaussian function, which has a low precision at the pixel level. In contrast, Mou and Zhu (2018) sought an instance segmentation of vehicles with pixel-level accuracy, where cars appear well delimited in a distinct physical instance. In this context, a severe problem is the differentiation of vehicles in contact that agglutinated in a single instance. The solution proposed by the authors was to establish an architecture that subdivided the central vehicle regions and their limits instead of treating the vehicle problem as a single unit. Reksten and Salberg (2021) recently used the Mask R-CNN with an image normalization strategy to suit different environments and an accurate road mask to filter driving vehicles from those parked.

Other studies combine a prior segmentation followed by vehicle detection. Audebert et al. (2017a) used the deep-learning-based segment-before-detect method containing three steps: (a) semantic segmentation using a fully convolutional network to infer pixel-level class masks; (b) vehicle detection by regressing the bounding boxes of connected components; and (c) object-level classification using CNN architectures (LeNet, AlexNet, and VGG-16). Yu et al. (2019) developed a convolutional capsule network with the following steps: (a) superpixel segmented, (2) labeling patches into vehicles or background using convolutional capsule network, and (3) non-maximum suppression to eliminate repetitive detections. Tao et al. (2019) performed a scene classification with deep learning followed by different vehicle detectors and post-processing rules according to the scene context.

Table (5.2) Related works using object detection algorithms, considering the method and data type. The data types are separated into seven categories: (1) satellite, (2) aerial, (3) UAV, (4) ultrahigh-resolution UAV, (5) Google Earth (GE), (6) Cameras at the top of the building, and (7) several. Acronyms for the methods: Residual Feature Aggregation (RFA), Generative Adversarial Network (GAN), and You Only Look Once (YOLO).

Paper	Method	Data
Chen et al. (2014)	Hybrid Deep Convolutional Neural Network (HDNN)	5
Qu et al. (2016)	Two-step detection: BING to extract region proposals and feature extraction for classification with CNN	1
Deng et al. (2017)	Two CNNs: AVPN to predict bounding boxes of the targets, and VALN for inferring type and orientation.	2
Tang et al. (2017)	An improved vehicle detection method based on Faster R-CNN.	3
Xu et al. (2017)	Vehicle detection using the Faster R-CNN	3
Zhong et al. (2017)	Method based on Cascaded Convolutional Neural Networks	2
Koga et al. (2018)	Hard Example Mining (HEM) to the Stochastic Gradient Descent training of a CNN classifier.	7
Liu et al. (2018d)	Real-Time Ground Vehicle Detection based on CNN.	3
Zhu et al. (2018)	Development of the Deep Vehicle Counting Framework based on Enhanced-SSD	4
Benjdira et al. (2019)	Comparison between YOLOv3 (best model) and Faster R-CNN	3
Chen et al. (2019)	Detection model based on two CNNs that adopt the VGG-16 model	2
Gao et al. (2019)	EOVNet (Earth observation image-based vehicle detection network), a modified Faster R-CNN.	7
Ji et al. (2019a)	Improved Faster R-CNN with Multiscale Feature Fusion and Homography Augmentation	7
Li et al. (2019b)	R3-Net a deep network for multi-oriented vehicle detection	2
Shen et al. (2019)	Detection algorithm based on Faster R-CNN	2
Sommer et al. (2019)	Systematic investigation of the Fast R-CNN and Faster R-CNN in vehicle detection	2
Wang et al. (2019)	YOLOv3, vehicle tracking using deep appearance features, and Kalman filtering for motion estimation	3
Xi et al. (2019)	Model based on multi-task cost-sensitive-convolutional neural network (MTCS-CNN)	6
Yang et al. (2019)	Novel double focal loss convolutional neural network (DFLCNN)	2
Zhang and Zhu (2019)	Improved YOLOv3 using a sloping bounding box attached to the angle of the target vehicles	2
Guo et al. (2020b)	Orientation-aware feature fusion single-stage detection (OAFF-SSD)	3
Li et al. (2020a)	Detection model for different scales using CNN and proposition of an Outlier-Aware Non-Maximum Suppression.	3
Ham et al. (2020)	Comparison among faster R-CNN, R-FCN, and SSD (Best model)	3
Jiang et al. (2020)	Optimized DL model considering feature extraction, object detection, and non-maximum suppression.	7
Mandal et al. (2020)	Small-Sized Vehicle Detection Network (AVDNet) (one-stage vehicle detection network)	2
Ophoff et al. (2020)	Comparison among four object detection networks: D-YOLO (best model), YOLOV2, YOLOV3, and YOLT	1
Stuparu et al. (2020)	Vehicle detection based on RetinaNet architecture	1
Tan et al. (2020)	Model based on Alexnet network (classification) and Faster R-CNN (target detection)	1
Wang and Gu (2020)	Faster R-CNN with a improved feature-balanced pyramid network (FBPN)	2
Ammar et al. (2021)	Comparison among YOLOv3, YOLOv4 (best models), and Faster R-CNN	3
Bashir and Wang (2021)	Super-resolution cyclic GAN with RFA and YOLO as the detection network (SRCGAN-RFA-YOLO)	1, 2
Javadi et al. (2021)	Modified YOLOv3 and fcNN using 3D features in cascade.	2
Shen et al. (2021)	Method using the lightweight feature extraction network with the Faster R-CNN	2
Shi et al. (2021)	Orientation-Aware Vehicle Detection with an Anchor-Free Object Detection approach	2

5.3 Material and methods

5.3.1 Study area and image acquisition

The entire city of Brasilia was the study area (Figure 5.2). Large regions with many mapped look-alike features and different scenarios favor learning DL models. The image has 57,856 x 42,496 spatial dimensions, and 0.24-meter resolution obtained by the Infraestrutura de Dados Espaciais do Distrito Federal (IDE/DF) (`https://www.geoportal.seduh.df.gov.br/geoportal/`, accessed on January 8, 2022). In this scenario, a car has approximately 20 (length) x 10 (width) pixel dimensions.

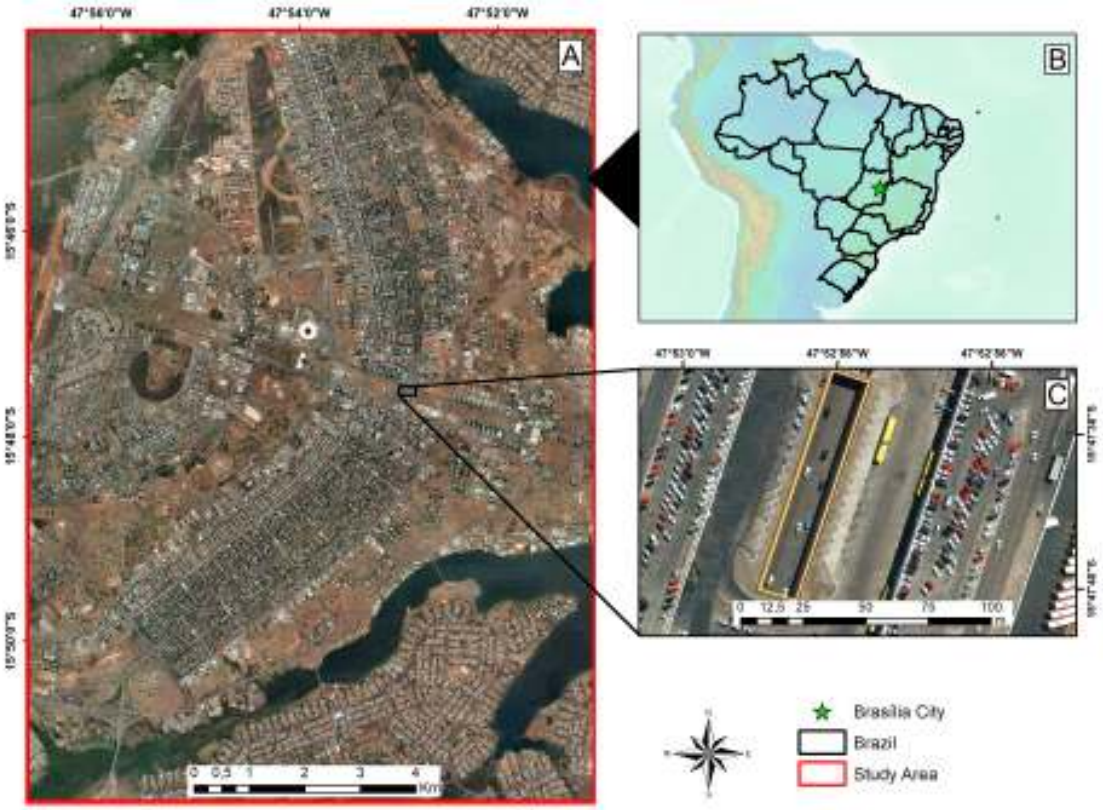

Figure (5.2) Study area.

5.3.2 Semi-supervised iterative learning

Manually identifying all the cars in a city is very time-consuming. So, the solution is to seek alternatives to automate the generation of datasets correctly. For example, if a very good annotator took five seconds to label a single car, it would take over 200 hours to label 150,000 vehicles. Thus, we proposed a novel semi-supervised approach using Geographic Information System (GIS) data to increase operability (Figure 5.3). Briefly, the method consists in labeling a portion of the image for training the model and then using the model to classify the entire 57,856 x 42,496-pixel image. Then, we converted the predictions into the shapefile format easily

edited in ArcMap, corrected the areas that present the most errors, and included them in the training data.

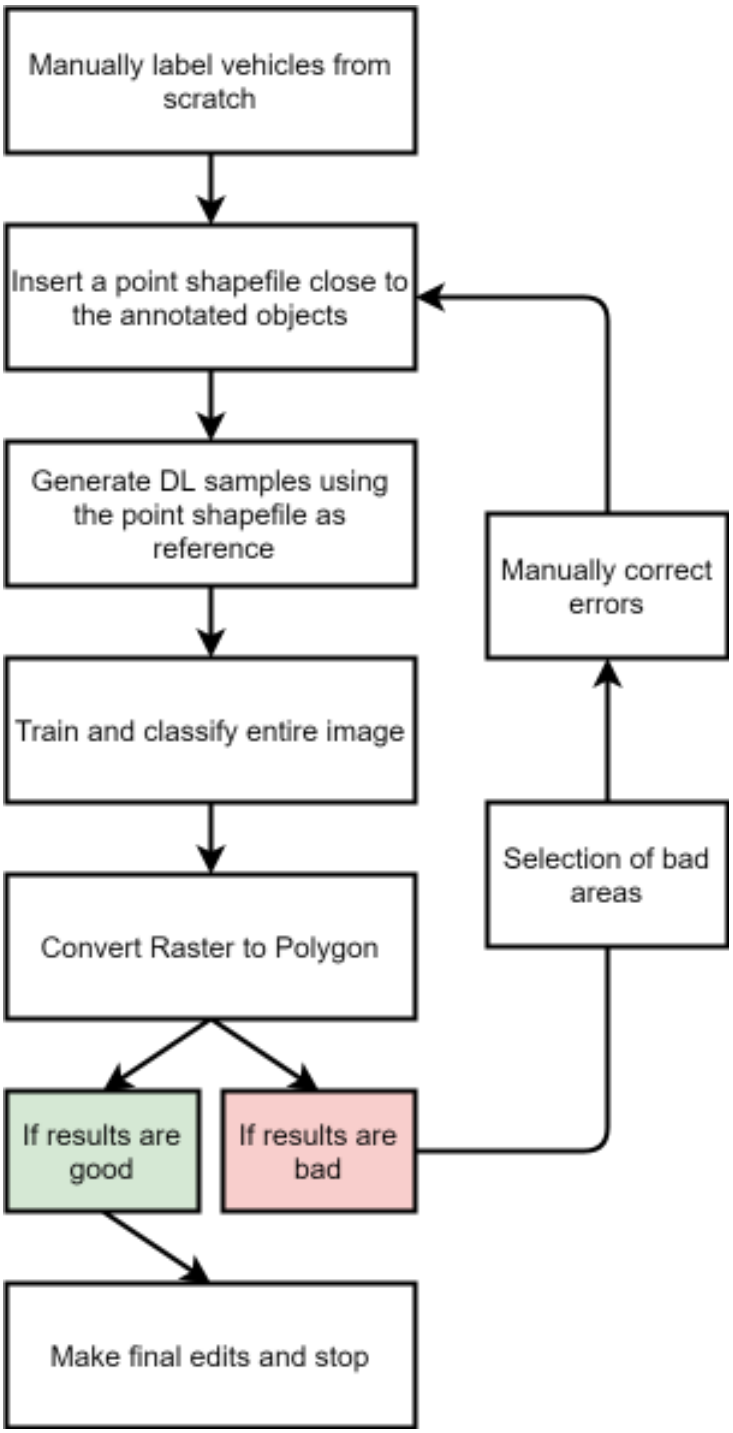

Figure (5.3) Proposed semi-supervised pipeline.

The proposed procedure to increase the training database reconciles incremental and cumulative learning, selecting samples that improve model performance. An effective database expansion design aims to achieve greater incremental accuracy in subsequent predictions. The procedure is cumulative, using the entire set of labeled samples present in each step. The seg-

mentation model increases its performance until the accuracy values do not vary significantly, i.e., the decrease in the incremental accuracy is due to the depletion of informative data.

Ground truth

The manual annotations and corrections used the ArcMap software, considering a polygon shapefile for each vehicle since it is much easier to manipulate when compared to raster (mask) data. We applied a 1-pixel buffer (0.24 meters in the corresponding image) with a negative distance to generate the borders inside the polygon features. The first training procedure used training samples made from scratch. Subsequent iterations used the DL predictions as the primary raw data, with corrections for the areas with the most errors. The number of verified and corrected areas increases after each iteration using the semi-supervised approach, increasing the dataset.

Deep learning sample generator software

The capture of DL samples must be in strategic areas. The present research proposed a novel method for selecting samples using the Point shapefile. This procedure allows choosing critical points where wrong predictions become part of new training after correction, quickly improving the model's detection capacity with much less laborious work. The developed DL sample generator from Point Shapefiles became a module in the Abilius Software program that receives three inputs: (1) the original image, (2) the ground truth image, and (3) the point shapefiles. The program requires inputs in the same projection, and the user may choose the size of the image tiles generated. The software uses the point shapefile to center the image tiles and crops the image and its corresponding ground truth image. This software outputs the annotations for instance segmentation, considering the COCO annotation format (Lin et al., 2014), which is compatible with Region CNN methods (Girshick et al., 2016), such as the Mask-RCNN (He et al., 2020) and similar methods. Using point shapefiles also enables the user to generate samples close to each other, a powerful augmentation technique.

Deep learning approach

Usually, region-based instance segmentation underperforms on small objects, and semantic segmentation does not present distinct classification for different instances, unable to differentiate adjacent vehicles. The conversion of a conventional semantic segmentation model to a polygon shapefile with touching vehicles (Figure 4A) acquires a single polygon. Semantic segmentation models are the most used among the remote sensing community, mainly because of the good per-pixel results and simplicity of models and annotation formats. To solve this problem, we adopted a similar solution proposed by Mou and Zhu (2018). Instead of multitasking learning, we adopted a multiclass learning procedure in which the contour class competes against the vehicle class.

The model output subdivides the vehicle into two parts (edge and interior) (Figure 5.4B). Deleting the edges isolates the individual vehicles, and all previously touching cars will be at least 2 pixels apart from each other. The next step is to develop a function to attribute a

different value to each vehicle. This proposed method generates a list with all contours, using the OpenCV function (findContours) (Bradski and Kaehler, 2008), and iteratively converts the contours to a mask attributing different values from 1 to N, being N the total number of distinct vehicles (Figure 5.4C). Aiming to optimize computational resources, we adapted the polygon2mask function from the scikit-image package (van der Walt et al., 2014) that generates an array with zeros every time it is called, which is costly due to the enormous image dimensions. Thus, we only create an array with zeros once. In each iteration, we attribute different values to the generated mask (one object at a time), guaranteeing distinct values for each vehicle.

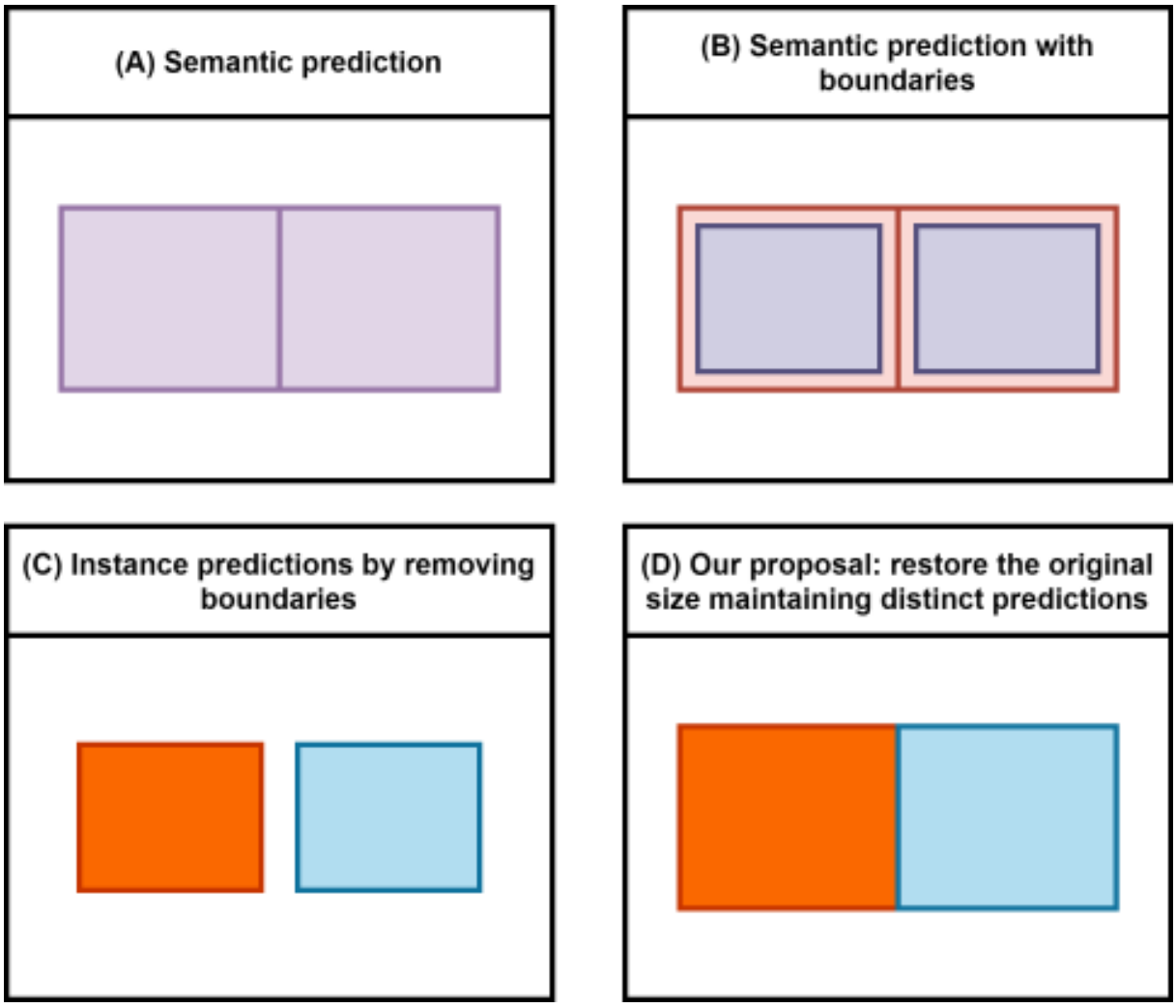

Figure (5.4) Theoretical outputs from semantic segmentation algorithms, in which A is a normal semantic segmentation strategy, B is segmentation with boundaries, C is instance segmentation by removing the boundaries, and D is our proposed solution to restore the correct size maintaining distinct predictions.

Now, the predictions are distinct for each object. However, since the objects are small, a 1-pixel error at the edges is considerable and not as precise. The edge restoration uses the instance array as the input. The first step is to apply 1-pixel padding in the entire image. Then we make eight copies of the original array dislocated in different directions: (1) up, (2) down, (3)

left, (4) right, (5) upright, (6) up-left, (7) down-right, and (8) down-left. Then we sum all arrays considering only pixels with zero value and remove the initial padding (recovering the images original shape). This procedure enlarges the object edges, independent of the object orientation, resulting in the same semantic information (Figure 5.4A), but with different instances for each object (Figure 5.4D).

Despite the variety of semantic segmentation models, the present study used a single combination throughout the iterative learning process since the primary goal is not to develop a new DL architecture but to make an efficient procedure for large areas per-pixel vehicle detection separating different instances. The configuration used the Semantic Segmentation Models repository (Yakubovskiy, 2020) and considered the U-net architecture Ronneberger et al. (2015b) with the Efficient-net-B7 backbone (Tan and Le, 2019b). Nevertheless, to present a more robust comparison, we evaluated the DeepLabv3+ (Chen et al., 2017), Pyramid Scene Parsing Network (PSPNet) (Zhao et al., 2017), Feature Pyramid Network (FPN) (Lin et al., 2017), and LinkNet (Chaurasia and Culurciello, 2017) on final generated dataset, all of which using the Efficient-net-B7 backbone.

The hyperparameters were the same for all training iterations: (a) 300 epochs, (b) Adam optimizer, and (c) batch size of five. The method considered the cross-entropy loss function with weights (0.1 for background, 0.6 for vehicles, 0.3 for the contour) and 15% of the images as validation, saving the model with the lowest cross-entropy loss. The dataset expansion used two augmentation strategies: the random horizontal and vertical flip, both with probabilities of 50%.

We compared the proposed method with the Mask-RCNN model (He et al., 2020) to evaluate the differences between a box-free method (ours) and a box-based method. In this context, the Detectron2 software is open source Wu et al. (2019), being one of the most widely used in instance segmentation. It is important to state that there are limitations in comparing box-free and box-based methods because: (1) the hyperparameters are different, (2) the models are different (both architectures and backbones), (3) the data format is different (e.g., instance segmentation models require data in the COCO annotation format). The proposed annotation tool simultaneously provides semantic segmentation ground truth and COCO annotations for compatibility with box-based methods.

Three backbone configurations were tested (ResNeXt-101 (Xie et al., 2017), ResNet-101 (He et al., 2016), and ResNet-50), all of which presents pre-trained weights, which speeds the training process. For box-based methods, a very substantial augmentation includes scaling the image dimensions, which increases the number of pixels for the object class, increasing results. In this regard, we considered two scenarios. The first considered the original image dimensions (256x256), and the second scenario scaled the image to 1024x1024-pixel dimension. This augmentation strategy is much harder for semantic segmentation models (using our computer configurations, requiring a more robust GPU) since the computational cost would increase substantially, running out of memory. In contrast, the instance segmentation models allow this strategy since the segmentation masks are performed only for the proposed boxes. Despite the differences, the comparison is valid to understand if our proposed method is better at pixel-level

accuracy, even using augmentations for the box-based instance segmentation methods that are not valid for our proposed method. In both cases, we used random horizontal and vertical flips. The training used 10,000 iterations, two images per batch, and the other parameters as default.

Large image classification

The dimensions of the training images are 256x256, which is smaller than the entire image. Thus, we considered a sliding window approach with a 128-pixel stride to classify the whole image. The stride size smaller than the image dimensions results in overlapping pixels. A traditional way is to take the mean average among the overlapping pixels. This approach reduces errors at the borders of the frames, exemplified in recent works (Costa et al., 2021; da Costa et al., 2021b; de Albuquerque et al., 2020b). A drawback of using this method is the computational cost. The time to classify an image increases non-linear when reducing the stride value. Since our image presents large dimensions, we did not consider smaller stride values.

5.3.3 Model evaluation

The model evaluation considered a test set of 50 images with 256x256-pixel dimensions (same as dimensions for training and validation), and three independent testing areas (Figure 5.5), considering different difficulty scenarios. The first considered areas with no occlusion and significant difficulties for the cars (Figure 5.5A), with 2560x2560-pixel dimensions. The second scenario is a parking lot with many crowded vehicles (Figure 5.5B) with 2304x2304-pixel dimensions. The third scenario cover residential areas with a building generating shadow and regions of occlusion (Figure 5.5C) with 1560x1560-pixel dimensions. The semantic segmentation of the entire test area used a sliding window with 128-pixel steps. Meanwhile, the instance segmentation (Mask-RCNN) of the testing areas used the mosaic method developed by Carvalho et al. (2021).

The accuracy analysis compares the predicted results and the ground truth data in supervised learning tasks. The confusion matrix is a standard structure for all tasks, yielding four possible outcomes: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). For semantic segmentation tasks, the confusion matrix analysis is per pixel. There are many possible metrics such as overall accuracy, precision, recall, and f-score, among others. Since we aim to evaluate how the metrics improve iteratively, we chose the Intersection over Union (IoU), widely adopted as one of the most critical semantic segmentation metrics. The IoU is given by:

$$IoU = \frac{|A \cap B|}{|A \cup B|} = \frac{TP}{TP + FP + FN} \tag{5.1}$$

In which $A \cap B$ is the area of intersection and $A \cup B$ is the area of union. The analysis considered: (a) IoU for the test set and the three testing areas (considering the proposed expanded border algorithm and without considering the borders) at each iteration; and (b) per-object metrics in the testing areas (T1, T2, and T3). The object analysis had four classifications: (1) correct predictions, (2) partial predictions, (3) false positives, and (4) false negatives.

Figure (5.5) Zoom from the three separate testing areas A, B, and C.

5.4 Results

5.4.1 Training iterations

The final version of the dataset used a total of five iterations. The total number of point shapefiles was 1066, with training samples in various scenarios (Figure 5.6). Each iteration considered point shapefiles in areas where the errors did not disappear in previous iterations (to see if the mistakes disappeared). Still, at each iteration, the concentration of points had different focuses. For example, the second training focused on eliminating look-alike features, which already give a good boost in performance metrics, with an easy correct the error, since we only need to delete some polygons. The fourth training had the minimum number of points since the areas required more corrections (e.g., parking lots), being more laborious. The proposed procedure effectively uses the results of the DL model in repeated corrections of pseudo-labels. Gradually, the predictions become more reliable, minimizing errors and manual correction labor in each interaction.

5.4.2 Metrics

Pixel metrics

Table 5.3 lists the results for IoU on the four separate testing sets (Test Area 1, Test Area 2, Test Area 3, and Test Set), considering each training step. There is an evident rise in the metrics when increasing the number of training samples on the same independent test areas. Test Area 1 (T1) had the highest results, and it is indeed the easiest since there are no shadows and occluded cars. Test area 2 (T2) has a parking lot with many crowded vehicles, presenting more errors. Test Area 3 (T3) has many regions with shadows, and partial vehicles had the lowest IoU, bringing to light the difficulty in some areas, even for human specialists. The test set has fifty 256x256 samples all around the city, with varying difficulty levels. The IoU of the test set is approximately the average of the distinct testing areas (81.88).

Table (5.3) IoU results for our proposed method in the BSB vehicle dataset considering the expanded (exp.) border algorithm and not considering the borders, for each train iteration.

Train #	Type	T1	T2	T3	Test Set
1	No border	63.19	63.67	51.97	52.60
	Exp. Border	80.80	77.23	66.89	66.03
2	No border	64.41	64.65	54.41	63.52
	Exp. Border	86.73	79.94	74.75	80.39
3	No border	60.40	62.27	52.43	61.49
	Exp. Border	87.69	82.31	75.95	80.73
4	No border	62.83	62.39	55.81	63.39
	Exp. Border	88.03	81.98	78.44	81.06
5	No border	63.98	64.13	56.24	64.51
	Exp. Border	88.37	81.31	77.10	82.45

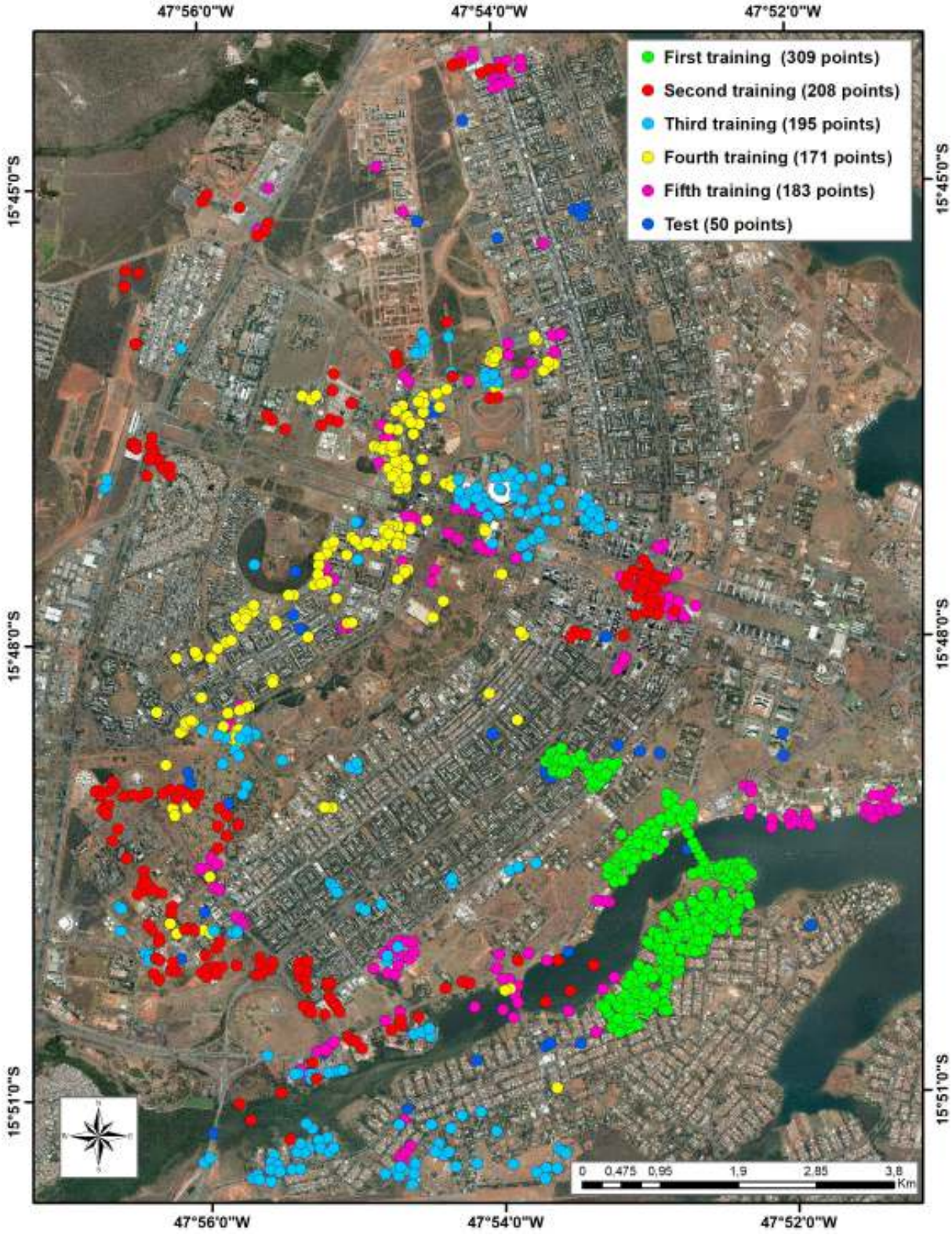

Figure (5.6) Study area with the Point Shapefiles (training points) used in each training, in which the training is cumulative.

Table lists the results considering different architectures using the Efficient-net-B7 backbone. For all models, the same behavior was still present, in which the expanded border algorithm had a higher value than without using the borders, showing that the method is not dependent on the model architecture used, but on the preparation of data. The PSPNet was by far the worst model, and the difference between the expanding border algorithm and without the borders was the lowest, showing that better models enhance the proposed algorithm even more. The DLv3+, LinkNet, and FPN presented slightly worse results than the U-net, demonstrating that the U-net was the best choice for this problem.

Table (5.4) IoU results considering the DeepLabv3+, LinkNet, PSPNet, and FPN architectures considering the expanded (exp.) border algorithm, and not considering the borders.

Model	Type	T1	T2	T3	Test Set
DLv3+	No border	63.33	59.58	50.64	62.55
	Exp. Border	86.36	74.27	67.04	78.05
LinkNet	No border	66.47	63.50	53.73	64.93
	Exp. Border	86.78	79.33	70.31	81.31
PSPNet	No border	61.92	57.46	51.43	63.86
	Exp. Border	79.48	61.96	58.92	69.78
FPN	No border	62.83	62.39	55.81	63.10
	Exp. Border	88.03	81.98	78.44	78.26

When comparing the IoU using our growing border algorithm to recover initial values without considering the borders, the results are very distinct, with a difference greater than 15% in the IoU metric. Also, the metrics remain very similar even when increasing the number of training samples. A possible explanation is error compensation, not bringing insightful information on the testing data.

Figure 5.7 shows the semantic segmentation result, with and without borders. The visual results demonstrate that the proposed method expands vectorially 1 pixel on the edges, consisting of a fast process. Furthermore, the instances show an efficient separation. Figure 5.7B (second row) demonstrates that the traditional predictions would merge the vehicles into a single polygon if we have not differentiated them with the borders. Expanding edges on different instances retrieves the same semantic prediction information but with the distinction of the vehicles.

Table 5.5 lists the same testing areas but considers the Mask-RCNN algorithm. Region algorithms rely on some procedures to enhance the classification of small objects. The results show that using the Mask-RCNN with scaling the input image to 1024x1024 spatial dimensions (four times the original size) improves the results in more than 5% of IoU for all backbones. However, pixel metrics results are still far from the results using semantic segmentation architectures, in which the best model (ResNeXt-101) was more than 10% lower in IoU than the U-net model.

Per object metrics

Table 5.6 lists the per object metrics (Correct predictions, partial predictions, false negatives, and false positives) on the three separate testing areas (T1, T2, and T3), considering the best model (containing all training samples). T1 classified all objects, showing that vehicles without

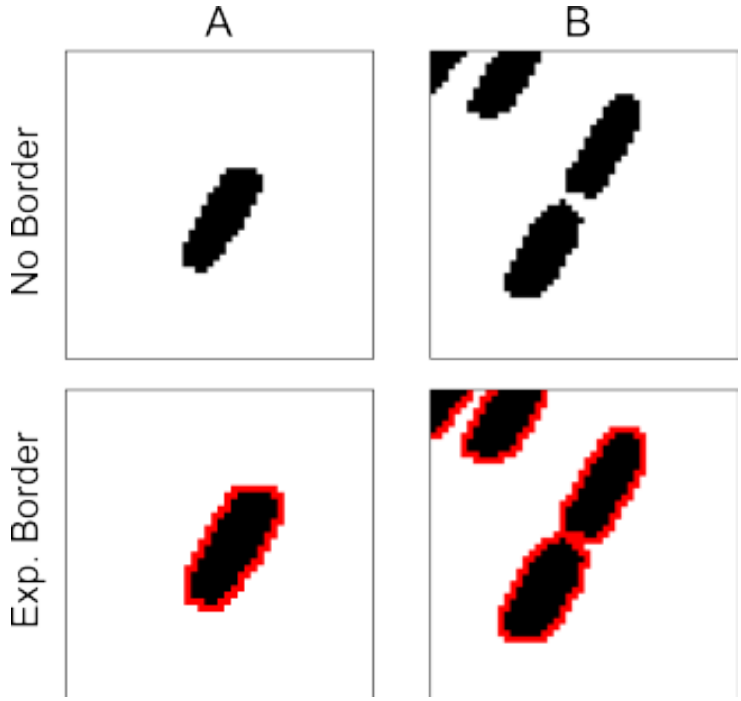

Figure (5.7) Representation of two examples considering the vehicles with no borders and with expanded borders, in which the borders are highlighted in red.

Table (5.5) IoU results for the Mask-RCNN with ResNeXt-101 (X-101), ResNet-101 (R-101), and ResNet-50 (R-50) backbones considering scaling augmentation (1024x1024 pixel dimensions) and without scaling augmentation (original 256x256 pixel dimensions) in the BSB vehicle dataset.

backbone	scaling	T1	T2	T3	Test Set
X-101	Yes	80.14	76.75	66.65	72.22
	No	75.41	63.88	55.17	67.06
R-101	Yes	80.54	72.32	64.93	72.02
	No	76.13	65.01	55.51	65.80
R-50	Yes	81.24	75.40	65.59	71.85
	No	79.40	66.59	55.32	66.49

shadows, occlusion, and crowded areas have very high precision. On the other hand, T3, with many shadow areas and occlusion, had the highest incidence of errors, with 21 false negatives and 25 false positives. Considering that there were 430 correct predictions, the accuracy was still greater than 90%.

Table (5.6) Per object metrics: Correct Predictions (CP), Partial Predictions (PP), False Negatives (FN), and False Positives (FP).

	T1	T2	T3
CP	89	395	430
PP	1	1	9
FN	0	5	21
FP	1	9	25

5.4.3 Semantic to instance segmentation results

Figure 5.8 shows three zoomed areas considering the traditional semantic segmentation method (first two rows) and our proposed box-free instance segmentation method. Both figures consider the same model. The first row (Figures 5.8A, 5.8B, and 5.8C) shows in yellow the merged cars, considering many vehicles in the same polygon, while the green cars were already independent even without our method. The second row (Figures 5.8A1, 5.8B1, and 5.8C1) shows the outlines of the polygons.

The third and fourth rows (Figures 5.8A2, 5.8B2, 5.8C2, 5.8A3, 5.8B3, and 5.8C3) show our proposed method considering the expanding border algorithm and separation into instance predictions. The fourth row shows cars in which each independent vector is represented by a different color, demonstrating that the method is efficient for separating vehicles in precise pixel classification. Besides, interpreting these results gets much more straightforward, estimating the sizes of the vehicles and more accurate counting.

5.4.4 Error analysis

Even though the results were very accurate, some regions contain limitations. The training procedure used many look-alikes features to train a better model. However, the number of look-alikes in a city is extensive, introducing some mistakes (Figure 5.9B, 5.9C, 5.9E, and 5.9F). Some crowded areas may raise some errors by joining two cars (Fig. 5.9A and 5.9D).

5.4.5 Final city-scale classification

The final city classification presented much fewer errors when compared to the first training. However, some errors were still present, as shown in the previous section. Figure 5.10 shows the final classified image with a manual correction using two GIS specialists. The data is publicly available with 122,567 vehicles (car, bus, truck, and boat) (Carvalho et al., 2022).

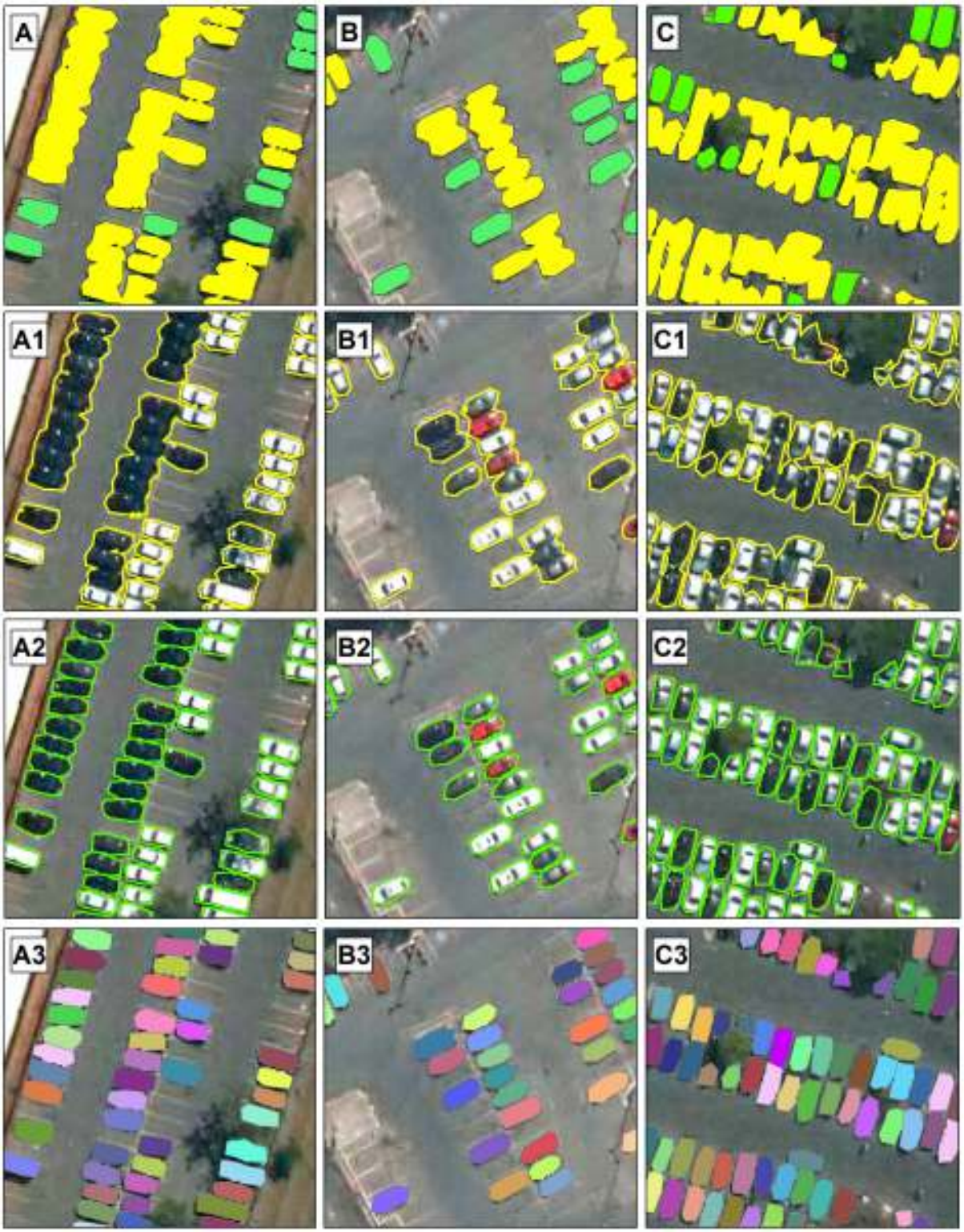

Figure (5.8) Visual comparison of the traditional semantic segmentation results without using the border procedure (first two rows), and the proposed method (last two rows).

Figure (5.9) Errors in the classification procedure, and errors present from the conversion from polygon to raster.

5.5 Discussion

5.5.1 Integration with GIS software

To the best of our knowledge, this research is the first to use semi-supervised iterative learning with GIS platform integration. We created a tool to generate the DL samples with corresponding ground truth data for semantic (PNG mask) and instance segmentation (COCO annotation format) to extract the best out of this method. A significant advantage of this method is understanding the misclassifications zones at each iteration, enabling choosing appropriate areas to continue the training with a substantial decrease in the laborious work. Generating training samples from point shapefiles allows a dataset augmentation by selecting points in strategic regions, enabling the acquisition of many samples in a limited space. This iterative approach stays in hand with Koga et al. (2018), supplying the algorithm with complex examples (e.g., look-alikes). Our method allows obtaining the exact points in which the algorithm confuses with hard examples, being able to supply those mistaken areas back to training, rapidly improving results.

The shapefile data is easy to manipulate, correct polygons, generate borders, and change classes, among others, reducing problems such as publicly available data with many errors in the ground truth data. Another great benefit is for end-users since the visualization of the data in those GIS platforms has many facilities, such as counting, choosing a specific area for analysis, and getting the average size of the objects. Therefore, DL and GIS systems may work as allies for generating better predictions in less time.

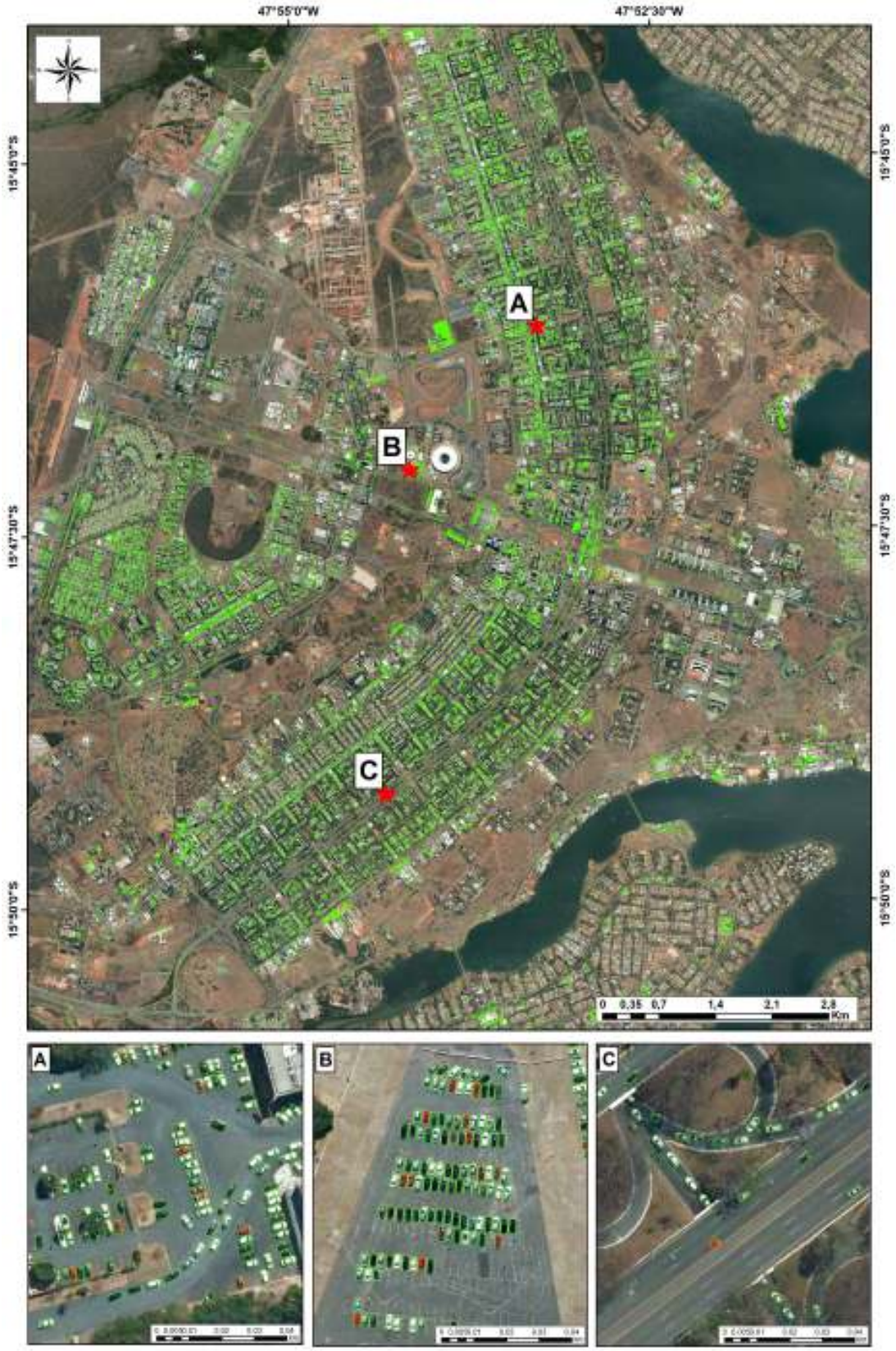

Figure (5.10) Final image classification with three zoomed areas A, B, and C.

5.5.2 Box-Free instance segmentation

The instance segmentation results for vehicle mapping pursue two goals: (a) high separability between objects and (b) high per-pixel precision. The traditional instance segmentation models are region-based methods with a segmentation branch like the Mask-RCNN. These box-based models have high object separability, but their pixel delimitation is lower than semantic segmentation models. Conversely, traditional semantic segmentation models cannot separate objects but have high per-pixel accuracy. Therefore, the present study seeks a different approach from the traditional methods of instance segmentation, adapting the configuration of the input data and the image post-processing procedures to obtain, from semantic segmentation methods, results of the instances with greater precision. We proposed a box-free instance segmentation method using semantic segmentation models with object separation by turning the interiors of the borders into distinct polygons and restoring the original object size. The border approach accurately isolates the objects, making it easy to attribute unique values to each vehicle using non-learning post-processing steps. Mou and Zhu (2018) had already introduced the usage of borders to separate instances. Even though the method is very interesting and effective, we incorporated the expanding border algorithm for more precise mapping. Our procedure uses a straightforward and fast vectorized approach to recover the 1-pixel at the borders of each object. In the literature, another proposal is by Tayara et al. (2018) which uses dots to represent each car with a Gaussian elliptical shape, but the segmentation masks for each vehicle are ellipticals differing from the car shapes, applied only for counting.

The proposed box-free instance segmentation method demonstrated a competitive and superior performance to the Mask R-CNN with different backbones and with and without image scaling. The application of image scaling is suitable for small objects (area $<32^2$ pixels) de Carvalho et al. (2021c), Tong et al. (2020), such as cars, increasing their detection capability. In tests restricted to Mask R-CNN, the best result considered ResNeXt-101 and image scaling to 1024x1024 pixel dimensions. However, the best Mask R-CNN result was lower than our method using U-net with Efficient-net-B7 backbone (72% versus 82%). Therefore, the proposed method generated high-quality maps with distinct polygons for each object and presented a good pixel-wise accuracy, demonstrating adequation for this task. Our proposed solution substitutes learning methods for object detection with non-learning methods, reducing the entire process's complexity. For example, the Mask-RCNN algorithm loss function is the sum of mask loss, classification loss, and box regression. In our proposed solution, we use a single loss function. We simplified the data preparation process, eliminating the bounding boxes or storing any information in JSON files for use with other software. The training procedure only requires the image with its corresponding mask (with the borders). A simple change in the data preparation process allows the application of instance segmentation with more precise pixel-wise results.

The step of restoring the original object size by expanding its borders by 1 pixel is a crucial factor in increasing accuracy metrics, reaching 15% more IoU than without the edge regardless of the architecture tested. Considering that the cars in the analyzed images have a dimension of 20x10 pixels, a perfect prediction only limited to the interior would reach only 72% IoU. The

better the model results, the greater the IoU differences between the result with and without the edge growth algorithm. (table IV). These results imply the procedure of augmenting the vehicle dataset using iterative learning, which must consider the features with reconstituted edges to delineate the objects better and compensate for errors. In addition, the evaluation of metrics per polygon in three test areas surpassed in all cases 90% in accuracy and recall. These results demonstrate an ideal scenario with good pixel mapping and the ability to distinguish different instances.

The large-area predictions using DL are an important topic that may be improved. Previous work shows that sliding windows with low step values correct errors at frame edges, improving results (Audebert et al., 2017a; Costa et al., 2021; da Costa et al., 2021b; de Albuquerque et al., 2020b). It takes about one hour to classify our entire study area (57,856 x 42,496-pixel dimensions) using a 128-pixel stride. Future studies may evaluate the usage of parallel computing to accelerate this process.

This method can be easily adapted to other remote sensing targets (e.g., airplanes, buildings, houses, swimming pools). There is no need to use the borders for some targets that do not appear crowded, such as swimming pools, since the predictions will already be separated when extracting the polygons from the predicted mask. There are two possibilities for multiple targets at once: (1) create a new class for each contour, and (2) create a single contour class for all classes. In both cases, the loss function would remain the same. However, depending on how balanced the classes are, it might be necessary to use weights on each class. This methodology could be enhanced to fulfill other segmentation tasks, such as panoptic segmentation (Kirillov et al., 2019) in remote sensing datasets, such as the BSB Aerial Dataset (de Carvalho et al., 2022b).

5.5.3 Vehicle dataset

A promising trend in Artificial Intelligence considers data-centric approaches, which consist of leveraging the data quality. In the present research, we aimed for a precise pixel-wise classification maintaining different instances for each object, being very relevant for vehicle studies since most vehicle datasets aim to use object detection models (only bounding boxes) (Azimi et al., 2021; Drouyer, 2020; Lin et al., 2020; Zeng et al., 2021). Some multi-class datasets also include vehicles (Xia et al., 2018a; Zamir et al., 2019). The iSAID dataset only comprises vehicles, for instance, segmentation tasks, with COCO annotation format annotations. Although object detection is very promising for counting vehicles, it requires adjustments (e.g., bounding box orientation) to obtain precise information (e.g., size), making the labeling procedure more complex. To obtain pixel information about the cars to generate a map, it is crucial to get the boundaries of each object. Our proposed method can obtain pixel-wise instance-level predictions with the same information required for a traditional semantic segmentation model, a box-free method. Furthermore, our proposed dataset stores polygonal data, facilitating additional adjustments such as dividing into more classes or refining labeled data.

Most vehicle studies use images with resolutions better than 20 cm. VAID (Lin et al., 2020) and VEDAI have the highest resolution (12.5cm) among the data sets. Our dataset has a pixel resolution of 0.24 meters, and the proposed method distinguished different instances at nearly twice the resolution of most datasets. The limitation of our dataset is that, for example, some distinguish sedans, which would be very difficult in our data. Therefore, our approach increases efficiency with a better resolution and is more suitable for separating into more classes (e.g., sedans, buses).

5.6 Conclusion

The present research presented three contributions: (a) a box-free instance segmentation method, (b) a semi-supervised iterative approach to generate a high-quality dataset, and (c) the BSB vehicle dataset. The proposed DL method shows better results when compared to the Mask-RCNN architecture with a pixel-wise IoU difference greater than 12%. We show that it is crucial to consider the borders for evaluating the pixel-wise mask, being very relevant to the proposed method to restore the objects original size. The semi-supervised iterative approach stabilized results in the fifth iteration, with a total of 1066 DL samples of 256x256 spatial dimensions. Our DL tool is a promising approach to generating datasets since it enables us to tackle strategic areas by inserting a point shapefile, significantly reducing laborious work. Finally, two specialists refined the BSB Vehicle Dataset containing more than 120 thousand unique vehicle polygons that are easily manipulated to other tasks.

The resolution in this research presents information very close to WorldView3 satellite imagery. Future research may consider the usage of more spectral bands in satellite data to enhance predictions. The results of our data are much better in situations without shadows and occlusion. For the generation of aerial imagery datasets, the researchers should consider training and evaluating the data in specific day periods with fewer shadows.

Chapter 6

Panoptic Segmentation

This chapter introduces the most novel image segmentation approach, which is still unexplored in the remote sensing field. This chapter introduces the first panoptic segmentation dataset that contains "things" and "stuff" classes and introduces software that converts GIS data into the panoptic segmentation format. The results from this chapter were published in Remote Sensing.

6.1 Presentation

Even though panoptic segmentation has excellent potential in remote sensing data, an image annotation that varies according to the segmentation task is a crucial step for its expansion. Semantic segmentation is the most straightforward approach, requiring the original image and its corresponding ground truth images. The instance segmentation has a more complicated annotation style, which requires the bounding box information, the class identification, and the polygons that constitute each object. A standard approach is to store all of this information in the Common Objects in Context (COCO) annotation format (Lin et al., 2014). Panoptic segmentation has the most complex and laborious format, requiring instance and semantic annotations. Therefore, the high complexity of panoptic annotations leads to a lack of remote sensing databases. Currently, panoptic segmentation algorithms are compatible with the standard COCO annotation format (Kirillov et al., 2019). A significant advantage of using the COCO annotation format is compatibility with state-of-the-art software. Nowadays, Detectron2 Wu et al. (2019) is one of the most advanced algorithms for instance and panoptic segmentation, and most research advances involve changes in the backbone structures, e.g., MobileNetV3 (Howard et al., 2019), EfficientPS (Mohan and Valada, 2021a), Res2Net (Gao et al., 2021). Therefore, this format enables vast methodological advances. However, a big challenge in the application of remote sensing is the adaptation of algorithms to its peculiarities, which include the image format (e.g., GeoTIFF and TIFF) and the multiple channels (e.g., multispectral and time series), which differ from the traditional Red, Green, and Blue (RGB) images used in other fields of computer vision (Carvalho et al., 2021).

The increase in complexity among DL methods (panoptic segmentation > instance segmentation > semantic segmentation) reflects the frequency of peer-reviewed articles across each DL approach (Figure 6.1). On the web of science and Scopus databases considering articles up to January 1, 2022, we evaluated four searches filtering by topic and only considering journal papers: (1) remote sensing AND semantic segmentation AND deep learning; (2) remote sensing AND instance segmentation AND deep learning; (3) remote sensing AND panoptic segmentation AND deep learning and (4) panoptic segmentation. Semantic segmentation is the most common approach using DL in remote sensing, while instance segmentation has significantly fewer papers. On the other hand, panoptic segmentation has only one research published in remote sensing (Hua et al., 2021), in which the authors used the DOTA (Xia et al., 2018b), UCAS-AOD (Liu et al., 2018a), and ISPRS-2D (`https://www2.isprs.org/commissions/comm2/wg4/benchmark/semantic-labeling/`, accessed on February 7, 2022) datasets, none of which are made for the panoptic segmentation task. We found two other studies, in which the first focuses on change detection in building footprints using bi-temporal images (Khoshboresh-Masouleh and Shah-Hosseini, 2021), and the second use for different crops (Garnot and Landrieu, 2021). Although both studies implement panoptic models, they do not use "stuff" categories apart from the background, which is very similar to an instance segmentation approach.

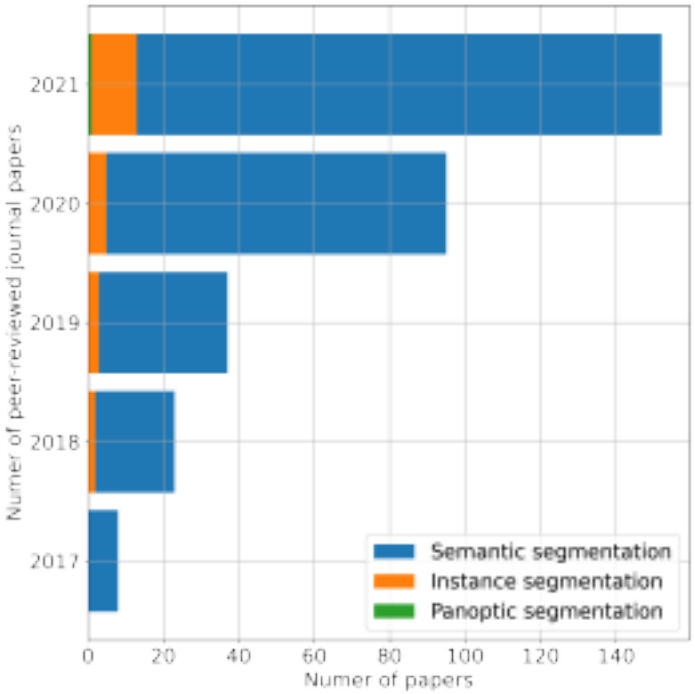

Figure (6.1) Temporal evolution of the number of articles in deep learning-based segmentation (semantic, instance, and panoptic segmentation) for the (A) Web of Science and (B) Scopus databases.

Even though the panoptic task is laborious, tools for easing the panoptic data preparation and integration with remote sensing peculiarities may present a significant breakthrough. The panoptic predictions retrieve countable objects and different backgrounds, guiding public poli-

cies and decision-making with complete information. The absence of remote sensing panoptic segmentation research alongside databases for this task represents a substantial gap. One of the notable drawbacks in the computer vision community regarding traditional images is the inference time, which exalts models like YOLACT and YOLACT++ Bolya et al. (2019, 2020) due to the ability to handle real data time, even compromising the accuracy metrics a little. This problem is less significant in remote sensing as the image acquisition frequency is days, weeks, or even months, making it preferable to use methods that return more information and higher accuracy rather than speed performance.

The advancements of DL tasks are strictly related to the disposition of large publicly available datasets, as in most computer vision problems, mainly after the ImageNet dataset (Deng et al., 2009). These publicly available datasets encourage researchers to develop new methods to achieve ever-increasing accuracy and, consequently, new strategies that drive scientific progress. This phenomenon occurs in all tasks, shown by progressively better accuracy results in benchmarked datasets. What makes the COCO and other large datasets attractive to test new algorithms is: (1) an extensive number of images; (2) a high number of classes; and (3) the variety of annotations for different tasks. However, up until now, the publicly available datasets for remote sensing are insufficient. First, there are no panoptic segmentation datasets. Second, the instance segmentation databases are usually monothematic, as many building footprints datasets such as the SpaceNet competition (Van Etten et al., 2018).

A good starting point for a large remote sensing dataset would include widely used and researched targets, and the urban setting and its components is a very hot topic with many applications: road extraction (Guo et al., 2020a; He et al., 2019a; Kestur et al., 2018; Lian and Huang, 2020; Mokhtarzade and Zoej, 2007; Senthilnath et al., 2020; Wu et al., 2021; Xu et al., 2018), building extraction Abdollahi et al. (2020); Bokhovkin and Burnaev (2019); Griffiths and Boehm (2019); Milosavljevic (2020); Rastogi et al. (2020); Sun et al. (2021); Yi et al. (2019), lake water bodies (Chen et al., 2018; Guo and Wang, 2020; Weng et al., 2020), vehicle detection (Ammour et al., 2017; Audebert et al., 2017b; Mou and Zhu, 2018), slum detection (Wurm et al., 2019), plastic detection (Jakovljevic et al., 2020), among others. Most studies address a single target at a time (e.g., road extraction, buildings), and panoptic segmentation would enable vast semantic information of images.

This study aims to solve these issues in panoptic segmentation for remote sensing images from data preparation up to implementation, presenting the following contributions:

1. **BSB Aerial Dataset**: a novel dataset with a high amount of data and commonly used thing and stuff classes in the remote sensing community, suitable for semantic, instance, and panoptic segmentation tasks.

2. **Data preparation pipeline and annotation software**: a method for preparing the ground truth data using commonly used Geographic Information Systems (GIS) tools (e.g., ArcMap) and an annotation converter software to store panoptic, instance, and semantic annotations in the COCO annotation format, that other researchers can apply in other datasets.

3. **Urban setting evaluation**: evaluation of semantic, instance, and panoptic segmentation metrics and evaluation of difficulties in the urban setting.

The remainder of this paper is organized as follows. The materials and methods section describes the study area, how the annotations were made, our proposed software, the Panoptic-Feature Pyramid Network (FPN) architecture, and the metrics used for evaluation. Next, the results section shows the outcomes and visual results. In the discussion, we present four topics of discussion retrieving the main contributions from this study (annotation tools, remote sensing datasets, difficulties in the urban setting, an overview of the panoptic segmentation task, and limitations and future works. Finally, we present the conclusions in the last section.

6.2 Material and Methods

The present research had the following methodology (Figure 6.2): (2.2) Data; (2.3) Conversion Software; (2.4) Panoptic Segmentation model; and (2.5) Model evaluation.

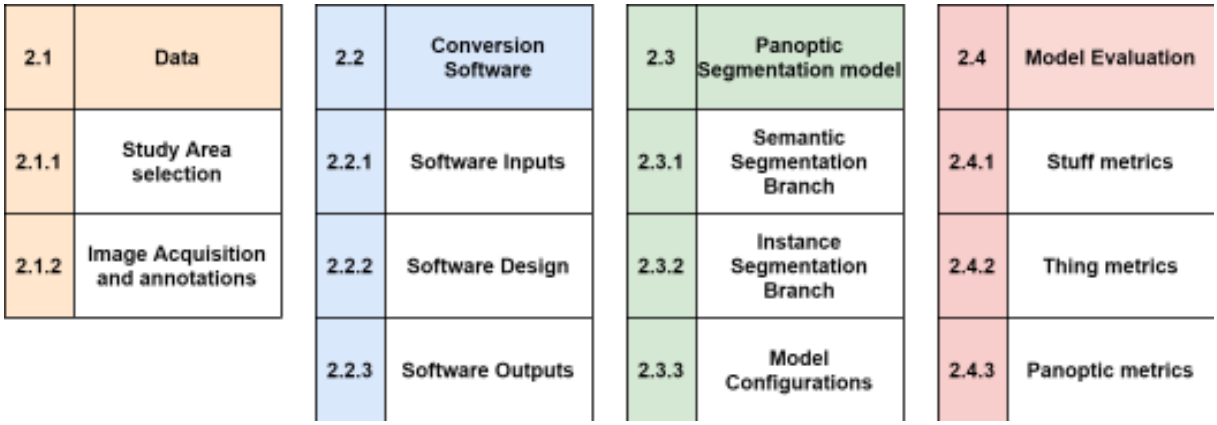

Figure (6.2) Methodological flowchart.

6.2.1 Data

Study Area Selection

The study area was the city of Brasília (Figure 6.3), the capital of Brazil. Brasília was built and inaugurated in 1960 by President Juscelino Kubitschek to transfer the capital of Rio de Janeiro (in the coastal zone) to the country's central region, aiming at modernization and integrated development of the nation. The capital's original urban project was designed by the urban planner and architect Lúcio Costa, who modeled the city around Paranoá Lake with a top-view appearance of an airplane. The urban plan includes housing and commerce sectors around a series of parallel avenues 13 km long, containing zones dedicated to schools, medical services, shopping areas, and other community facilities. In 1988, United Nations Educational, Scientific and Cultural Organization (UNESCO) declared the city a World Heritage Site.

The city presents suitable characteristics for DL tasks: (1) it is one of the few planned cities in the world presenting well-organized patterns, which eases the process of understanding each

Figure (6.3) Study area.

class; (2) the buildings are not high, which reduces occlusion and shadows errors due to the photographing angle; (3) the city contains organized portions of houses, buildings, and commerce, facilitating the annotation procedure; and (4) it has many socio-economical differences in many parts of the city, bringing information that might be useful to many other cities in the world. The city setting is very suitable for developing panoptic segmentation applications since it presents countable objects (e.g., cars and houses) and amorphous targets (e.g., vegetation and lake) that wouldn't be correctly represented by using only an instance or semantic segmentation approach.

Image Acquisition And Annotations

The aerial images present the RGB channels and spatial resolution of 0.24 meters over Brasilia cover an area of 79.40 km^2, obtained by the Infraestrutura de Dados Espaciais do Distrito Federal (IDE/DF) (`https://www.geoportal.seduh.df.gov.br/geoportal/`, accessed on February 7, 2022). We made vectorized annotations using the ArcMap software considering fourteen urban classes (three "stuff" and eleven "thing" categories). Table 6.1 lists the panoptic categories with their annotation pattern, and Figure 6.4 shows three examples from each class. The vehicles presented the most polygons (84,675), whereas the soccer fields had only 89. This imbalance

among the different categories is widespread due to the nature of the urban landscape, i.e., there are more cars than soccer fields in cities. The understanding of this imbalance is an essential topic for investigating DL algorithms in the city setting. Since there is high variability in the permeable areas, we made a more generalized class considering all types of natural lands and vegetation, being the class with the highest number of annotated pixels (803,782,026). The vehicle and boat polygons were obtained from de Carvalho et al. (2021e).

Table (6.1) Category, numeric label, thing/stuff, and the number of instances used in the BSB Aerial Dataset. The number of polygons in the stuff categories receives the '-' symbol since it is not relevant.

Category	**Label**	**Thing/Stuff**	**Polygons**	**Pixels**
Background	0	-	-	112,497,999
Street	1	Stuff	-	167,065,309
Permeable Area	2	Stuff	-	803,782,026
Lake	3	Stuff	-	117,979,347
Swimming pool	4	Thing	4,835	3,816,585
Harbor	5	Thing	121	214,970
Vehicle	6	Thing	84,675	11,458,709
Boat	7	Thing	548	189,115
Sports Court	8	Thing	613	3,899,848
Soccer Field	9	Thing	89	3,776,903
Com. Buiding	10	Thing	3,796	69,617,961
Res. Buiding	11	Thing	1,654	8,369,418
Com. Building Block	12	Thing	201	30,761,062
House	13	Thing	5,061	42,528,071
Small Construction	14	Thing	4,552	2,543,032

6.2.2 Conversion Software

DL methods require extensive collections of annotated images with different object classes for training and evaluation. Different open-sourced annotation software has been proposed, containing high-efficiency tools for the creation of polygons and bounding boxes, such as Labelme (Russell et al., 2008; Torralba et al., 2010), LabelImg (`https://github.com/tzutalin/labelImg`, accessed on February 7, 2022), Computer Vision Annotation Tool (CVAT) (Sekachev et al., 2019), RectLabel (`https://rectlabel.com`, accessed on February 7, 2022), Labelbox (`https://labelbox.com`), and Visual Object Tagging Tool (VoTT) (`https://github.com/microsoft/VoTT`, accessed on February 7, 2022).However, the elaboration of annotations in remote sensing differs from other computer vision procedures that use traditional photographic images (e.g., cellphone photos), containing some particularities, such as georeferencing, projection, multiple channels, and GeoTIFF files. There is a gap in specific annotation tools for remote sensing. In this context, a powerful solution for expanding the terrestrial truth database for DL is to take advantage of the extensive mapping information stored in a GIS database. GIS programs already have several editing, and manipulation tools developed and improved for geo-referenced data. Recently, a specific annotation tool for remote sensing is the LabelRS based on ArcGIS

Figure (6.4) Three examples of each class from the proposed BSB Aerial Dataset: (A1-3) street, (B1-3) permeable area, (C1-3) lake, (D1-3) swimming pool, (E1-3) harbor, (F1-3) vehicle, (G1-3) boat, (H1-3) sports court, (I1-3) soccer field, (J1-3) commercial building, (K1-3) residential building, (L1-3) commercial building block, (M1-3) house, and (N1-3) small construction.

(Li et al., 2021), considering semantic segmentation, object detection, and image classification. However, LabelRS is based on ArcPy scripts dependent on ArcGIS, not fully open-source, and does not operate with panoptic annotations.

The present study develops a module within the Abilius software that converts GIS vector data into COCO-compatible annotations widely used in DL algorithms (Figure 6.5) (`https://github.com/abilius-app/Panoptic-Generator`). The proposed framework generates samples from vector data in shape format to JavaScript Object Notation (JSON) files in the COCO annotation format, considering the three main segmentation tasks (semantic, instance, and panoptic). The use of GIS databases provides a practical way to expand the free community-

maintained datasets, minimizing the time-consuming and challenging process of manually generating large numbers of annotations for different classes of objects. The tool generates annotations for the three segmentation tasks in an end-to-end approach, in which the annotations are ready to use, requiring no intermediary process and reducing labor-intensive work. Besides, it is essential to note that the conversion from raster data to polygons may bring imprecision at a pixel level since points represent the polygons. This imprecision can be minimized by changing the approximation function for the polygon generation. However, when considering more points for each polygon, the computational power increases, and those approximation differences are imperceptible for the spatial resolution of our images. This tool was crucial to building the current dataset, but it also applies to other scenarios, since it just requires other researchers to follow our proposed pipeline using GIS software.

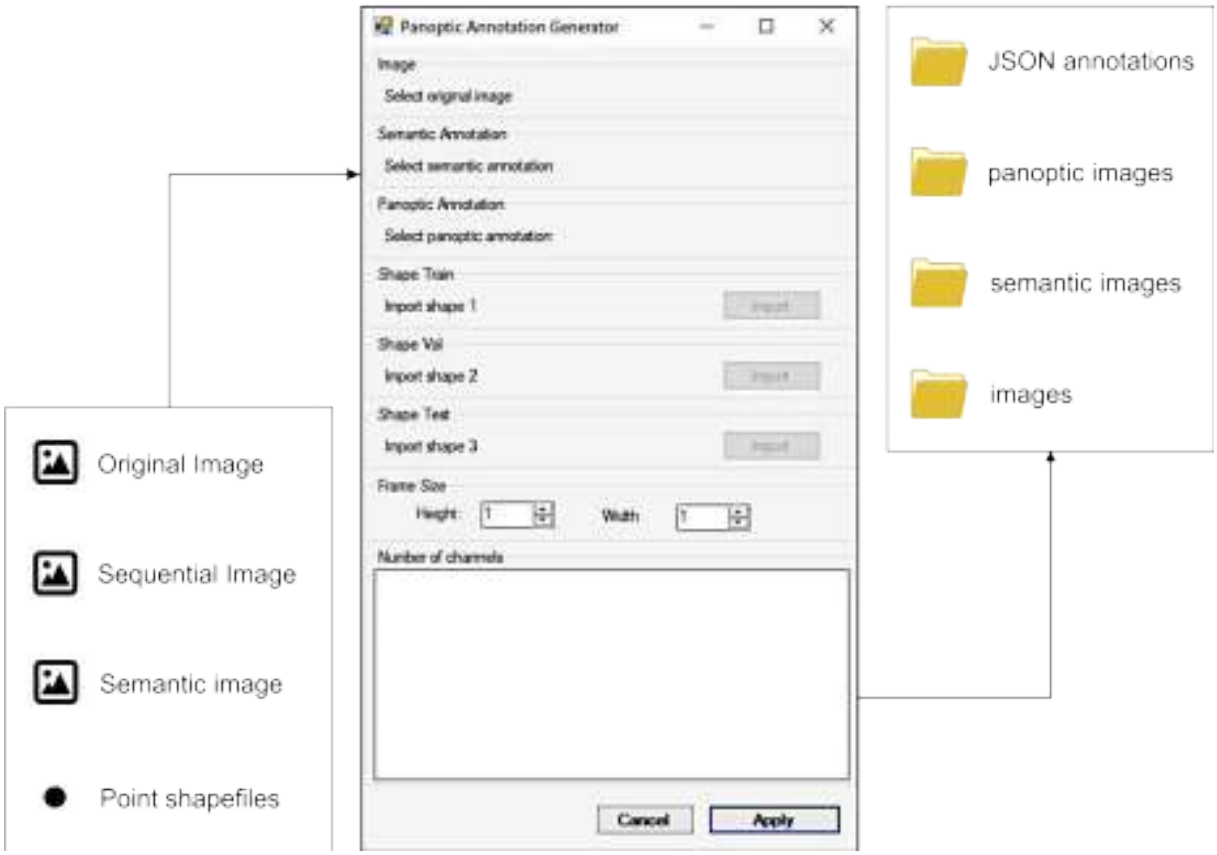

Figure (6.5) Flowchart of the proposed software to convert data into the panoptic format, including the inputs, design, and outputs.

Software Inputs

To automatically obtain the semantic, instance, and panoptic annotations, we proposed a novel pipeline with four inputs (considering the georeferenced images in the same system): (a) the original image (Figure 6.6A); (b) semantic image (Figure 6.6B); (c) sequential ground truth image (Figure 6.6C) (each thing object has a different value), and (d) the point shapefiles (Figure 6.6D). The class-agnostic image is a traditional semantic segmentation ground truth, in

which each class receives a unique label, easily achieved by converting from polygon to raster in GIS software. The sequential ground truth (which will become the panoptic images) requires a different value for each polygon that belongs to the thing categories. First, we grouped all the stuff classes since these classes do not need a unique identification. The subsequent thing classes receive a unique value for each polygon using sequential values in the attribute table.Point shapefiles play a crucial role in generating the DL samples since it uses the point location as the centroid of the frame. Our proposed method using point shapefiles provides the following benefits: (a) more control over the selected data in each set; (b) allows augmenting the training data by choosing points close to each other; and (c) in large images, there are areas with much less relevance, and the user may choose more significant regions to generate the dataset. Apart from the inputs, the user may choose other parameters such as spectral bands and spatial dimensions. Our study used the RGB channels (other applications might require more channels or less depending on the sensor) and 512x512-pixel dimensions.

Software Design

Given the raw inputs, the software must crop tiles in the given point shapefile areas. For each point shapefile, it crops all input images considering the point as the centroid, meaning that if the user chooses a tile size of 512x512, the frame will present a distance from the centroid of 256 pixels in the up, down, right, and left directions (resulting in a squared frame with 512x512 dimensions). Now, for each 512x512 tile, we must gather the image annotations semantic, instance, and panoptic segmentation tasks, given as follows:

- **Semantic segmentation annotation**: Pixel-wise classification of the entire image with the same spatial dimensions from the original image tiles. Usually, the background (i.e., unlabeled data) has a value of zero. Each class presents a unique value.

- **Instance segmentation annotation**: Each object requires a pixel-wise classification, bounding box, and class of each bounding box for each object. Since there is more information when compared to the semantic segmentation approach, most software adopts the COCO annotation format, e.g., Detectron2 Wu et al. (2019). For instance segmentation, the COCO annotation format uses a JSON file requiring for each object the: (a) identification, (b) image identification, (c) category identification (i.e., the label of the class), (d) segmentation (polygon coordinates), (e) area (total number of pixels), (f) bounding box (four coordinates) (`https://cocodataset.org/#format-data`, accessed on February 7, 2022).

- **Panoptic segmentation annotation**: The panoptic segmentation combines semantic and instance segmentation. It requires a folder with the semantic segmentation images in which all thing classes have zero value, and the instance segmentation JSON file and an additional panoptic segmentation JSON file. The panoptic JSON is very similar to the instance JSON, but considering an identifier named isthing, in which the thing category is one and stuff is zero.

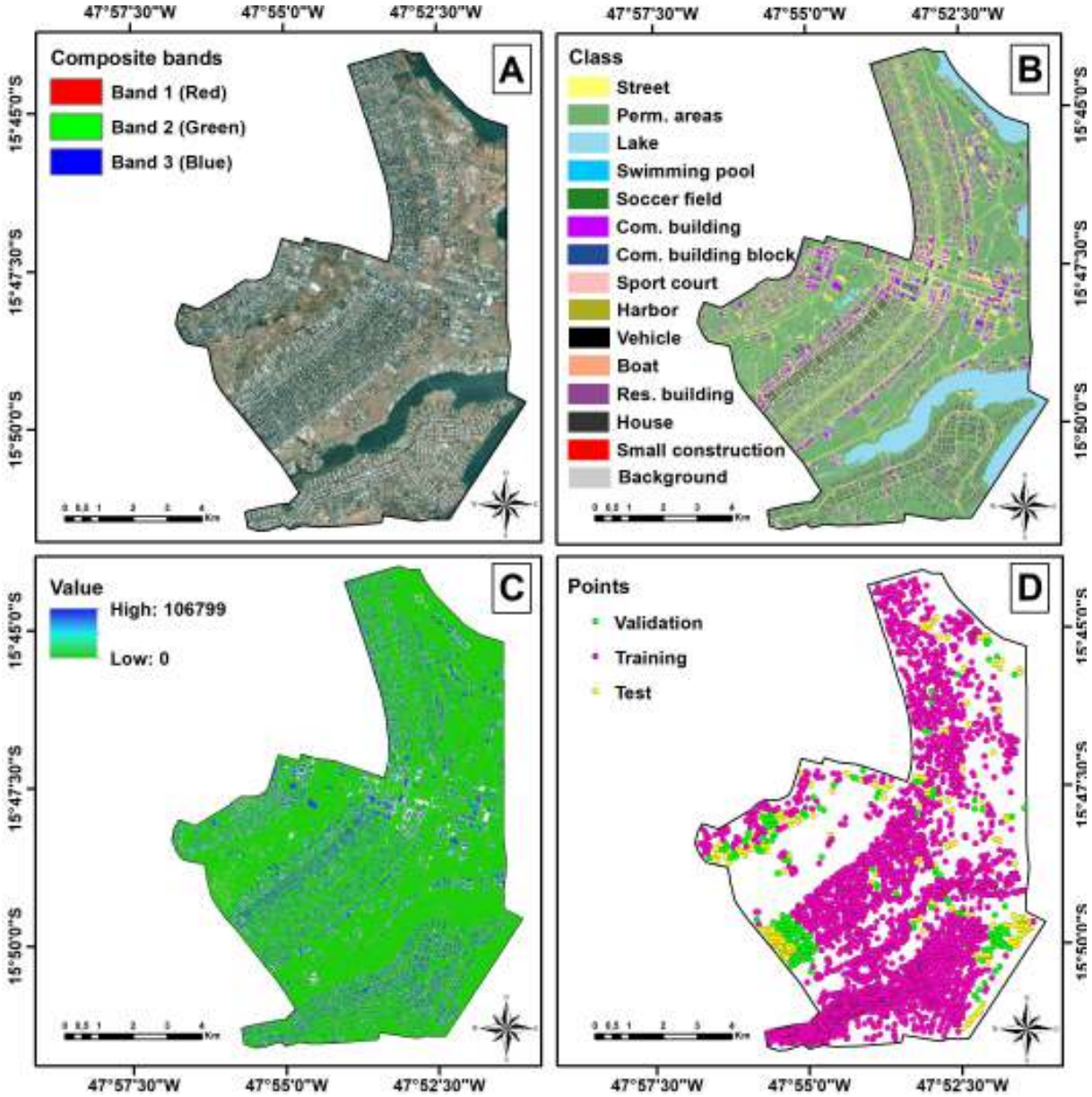

Figure (6.6) Inputs for the software in which (A) is the original image, (B) Semantic image, (C) sequential image, and (D) the point shapefiles for training, validation, and testing.

The semantic segmentation data is the most straightforward, and its output cropped tiles are already in the format to apply a semantic segmentation model. Nevertheless, the semantic image plays a crucial role in the instance and panoptic JSON construction. The parameters designed to build the COCO annotation JSONS for instance and panoptic segmentation were the following:

- **Image identification**: Each cropped tile receives an ascending numeration. For example, there are 3,000-point shapefiles in the training set, and the image identifications range from 1 to 3,000.

- **Segmentation**: We used the OpenCV C++ library for obtaining all contours in the sequential

image. The contour representation is in tuples (x and y). For each distinct value, the proposed software gathers all coordinates separately according to the COCO annotation specifications. The polygon information will only be stored in the instance segmentation JSON, but these coordinates will guide the subsequent bounding box process.

- **Bounding box**: Using the polygons obtained in the segmentation process enables the extraction of minimum and maximum points (in the horizontal and vertical directions). There are many possible ways to obtain the bounding box information using four coordinates. However, we used the top-left coordinates associated with the width and height.

- **Area**: We apply a loop to count the number of pixels of each different value on the sequential image.

- **Category identification**: This is where the segmentation image is so important. The sequential image does not contain any class information (only that each thing class has a different value). For each generated polygon, we extract the category value from the semantic image to use it as the category identification label.

- **Object identification**: This method is different for the instance and panoptic JSONS. In the instance JSON, the identification is a sequential ascending value (the last object in the last image will present the highest value, and the first object in the first image will present the lowest value), and it only considers the thing classes. In the panoptic JSON, the identification is the same as the object number in sequential order, and it considers thing and stuff classes.

Apart from these critical parameters, we did not consider the possibility of crowded objects (our data has all separate instances), so the is_crowd parameter is always zero. The user must specify which classes are stuff or things. The sequential input data is an image with single-channel TIFF format transformed in our software to a three-channel PNG image compatible with Detectron2 software, converting from a decimal number to base-256.

Software Outputs

The software outputs the images and annotations in a COCO dataset structure. The algorithm produces ten folders, an individual folder for annotations in JSON format, and three folders for each set of samples (training, validation, and testing) referring to the original image, panoramic annotations, and semantic annotations. In the training-validation-test split, the training set usually presents most of the data for the purpose of learning the specific task. However, the training set alone is not sufficient to build an effective model since, in many situations, the model overfits the data after a certain point. The validation set allows tracking the trained model performance on new data while still tuning hyperparameters. The test set is an independent set to evaluate performance. Table 6.2 lists the number of tiles in each set and the total number of instances. Our proposed conversion software allows overlapping image tiles, which may be valuable in the training data functioning as a data augmentation method. However, this would

lead to biased results if applied in the validation and testing sets. In this regard, we used the graphic Buffer analysis tool from the ArcMap software, considering the dimensions generating 512x512 squared buffers to verify that none of the sets were overlapping.

Table (6.2) Data split into the three sets with their respective number of images and instances, in which all images present 512x512x3 dimensions.

Set	Number of tiles	Number of instances
Training	3,000	102,971
Validation	200	9,070
Test	200	7,237

6.2.3 Panoptic Segmentation Model

With the annotations in the correct format, the next step was to use panoptic segmentation DL models. Panoptic segmentation networks aim to combine the semantic and instance results using a simple heuristic method (Kirillov et al., 2019). The model presents two branches: semantic segmentation (Figure 6.7B) and instance segmentation (Figure 6.7C). Figure 6.7 shows the Panoptic-FPN architecture, which use the FPN (Lin et al., 2017) as a common structure for both branches (Figure 6.7A. We considered two backbones, the ResNet-50 and ResNet-101.

Semantic Segmentation Module

Semantic segmentation models are the most used among the remote sensing community, mainly because of the good results and simplicity of models and annotation formats. There are a wide variety of architectures such as the U-net (Ronneberger et al., 2015b), Fully Convolutional Networks (FCN) (Zhang et al., 2018b), DeepLab (Chen et al., 2018). The semantic segmentation using the FPN presents some differences when compared to traditional encoder-decoder structures. FPN predictions with different scales (P2, P3, P4, P5) are resized to the input image spatial resolution by applying bilinear upsampling, in which the sampling rate is different for each prediction to obtain the same dimensions as shown in Figure 6.7B. The elements present in the things category all receive the same label (avoiding problems with the predictions from the instance segmentation branch).

Instance Segmentation Module

Instance segmentation had a significant breakthrough with the Mask-RCNN (He et al., 2020). This method relies on the extension of Faster-RCNN (Girshick, 2015), a detector with two stages: (a) Region Proposal Network (RPN); and (b) box regression and classification for each Region of Interest (ROI) from the RPN. However, aiming to perform pixel-wise segmentation, the Mask-RCNN added a segmentation branch on top of the Faster-RCNN architecture. First, the method applies the RPN on top of different scale predictions (e.g., P2, P3, P4, P5) and proposes several anchor boxes in more susceptible regions. Then, the ROI align procedure standardizes each bounding box dimension (avoiding quantization problems) as shown in Figure 6.7C. The last

step considers a binary segmentation mask for each object alongside the bounding box with its respective classification.

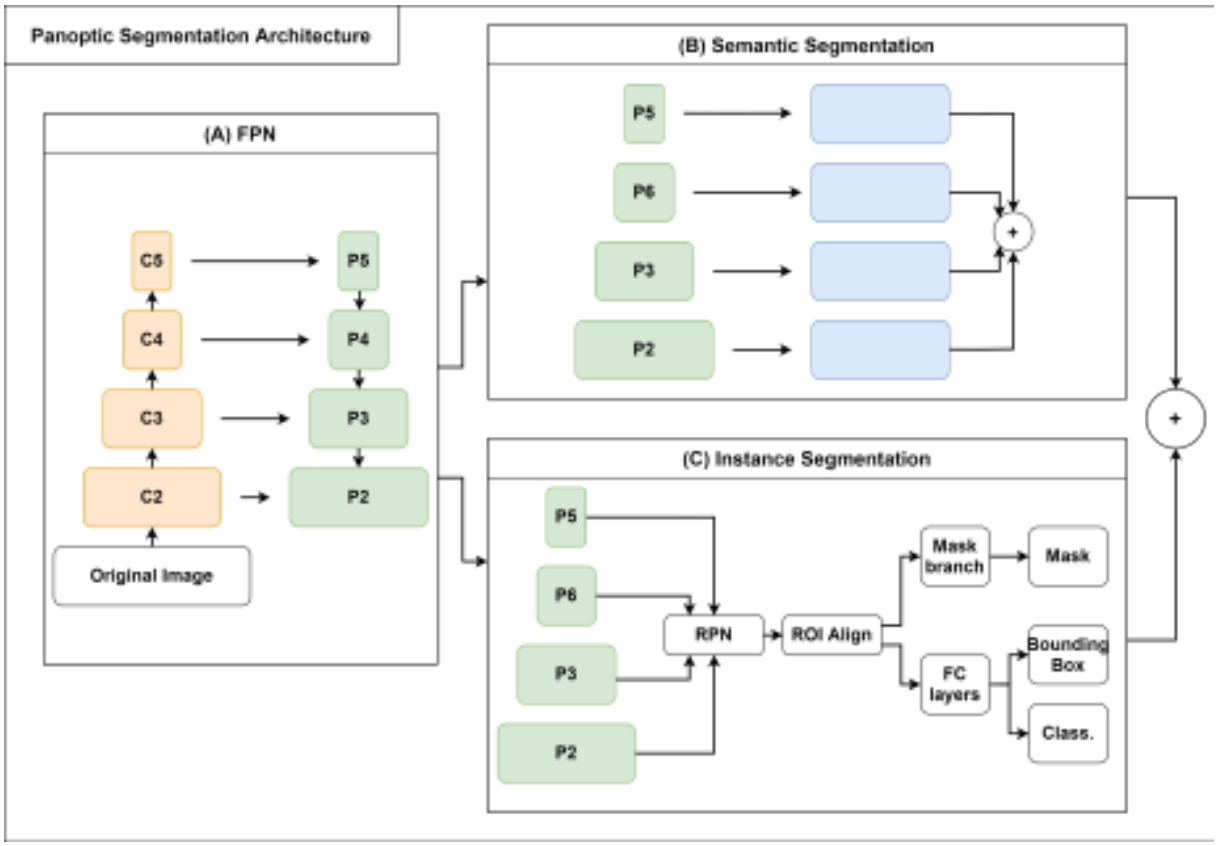

Figure (6.7) Simplified Architecture of the Panoptic Feature Pyramid Network (FPN), with its semantic segmentation (B) and instance segmentation (C) branches. The convolutions are represented by C2, C3, C4, and C5 and the predictions are represented by P2, P3, P4, and P5.

Model Configurations

The loss function for the Panoptic-FPN model is the combination of the semantic and instance segmentation losses. The instance segmentation encompasses the bounding box regression, classification, and mask losses. The semantic segmentation uses a traditional cross-entropy loss among the stuff categories and a class considering all thing categories together.

Regarding the model hyperparameters, we used: (a) stochastic gradient descent (SGD) optimizer, (b) learning rate of 0.0005, (c) 150,000 iterations, (d) five anchor boxes (with sizes 32, 64, 128, 256, and 512), (e) three aspect ratios (0.5, 1, 2), (f) one image per batch. We trained the model using ImageNet pre-trained weights and unfreezing all layers. Moreover, we evaluated the metrics on the validation set with a period of 1,000 iterations and saved the final model with the highest PQ metric. To avoid overfitting and increase performance (mainly on the small objects), we used three augmentation strategies: (a) random vertical flip (probability chance of 50%), (b) random horizontal flip (probability chance of 50%), and (c) resize the shortest edge with 640,

672, 704, 736, 768, and 800 possible sizes. The data processing used a computer containing an Intel i7 core and NVIDIA 2080 GPU with 11GB RAM.

6.2.4 Model Evaluation

In supervised learning tasks, the accuracy analysis compares the predicted results and the ground truth data. Each task has different ground truth data and, therefore, different evaluation metrics. However, the confusion matrix is a common structure for all tasks, yielding four possible results: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Sections 2.4.1, 2.4.2, and 2.4.3 explain the semantic, instance, and panoptic segmentation metrics, respectively.

Stuff Evaluation

For semantic segmentation tasks, the confusion matrix analysis is per pixel. The most straightforward metric is the pixel accuracy (pAcc):

$$pAcc = \frac{TP + TN}{TP + TN + FP + FN} \tag{6.1}$$

However, in many cases, the classes are imbalanced, bringing imprecise results. The mean pixel accuracy (mAcc) takes into consideration the number of pixels belonging to each class, performing a weighted average.

Apart from PA, the intersection over union (IoU) is the primary metric for many semantic segmentation studies, mainly because it penalizes the algorithm for FP and FN errors:

$$IoU = \frac{|A \cap B|}{|A \cup B|} = \frac{TP}{TP + FP + FN} \tag{6.2}$$

In which: $A \cap B$: the area of intersection; $A \cup B$: the area of union.

For a more general understanding of this metric, we may use the mean IoU (mIoU), which is the average IoU of all categories or the frequency weighted IoU (fwIoU) which is the weighted average of each IoU considering the frequency of each class.

Thing Evaluation

Instance segmentation metrics take into consideration both the bounding box predictions and the mask quality. The most common approach to instance segmentation problems uses standard COCO metrics (Bolya et al., 2020; Cai and Vasconcelos, 2018; Gao et al., 2021; He et al., 2020; Huang et al., 2019). The primary metric in evaluation is the average precision (AP) (Lin et al., 2014), also known as the area under the precision-recall curve:

$$AP = \int_0^1 Precision\,(Recall)\,dRecall, \tag{6.3}$$

in which:

$$Precision = \frac{TP}{TP + FP} \tag{6.4}$$

$$Recall = \frac{TP}{TP + FN} \tag{6.5}$$

The COCO AP metrics consider different IoU thresholds from 0.5 to 0.95 with 0.05 steps, which is useful to measure the quality of the bounding boxes compared to the original image. The secondary metrics consider specific IoU thresholds: AP_{50} and AP_{75}, which use IoU values of 0.5 and 0.75, respectively. Besides, the evaluation considers different sized objects (AP_S, AP_M, and AP_L): (1) small objects ($< 32^2$ pixels); (2) medium objects (32^2 pixels $<$ area $< 96^2$ pixels); and (3) large objects ($> 96^2$ pixels).

Panoptic Evaluation

The Panoptic Quality (PQ) is the primary metric for evaluating the Panoptic Segmentation task (Gao et al., 2021; Kirillov et al., 2019; Mohan and Valada, 2021a), and it is the current metric for the COCO panoptic task challenge, being defined by:

$$PQ = \frac{\sum_{(p,g)\in TP} IoU\,(pred, GT)}{|TP| + \frac{1}{2}\,|FP| + \frac{1}{2}\,|FN|} \tag{6.6}$$

In which p is the DL prediction, and g is the ground truth. The expression above is the multiplication of two metrics, the Segmentation Quality (SQ) and Recognition Quality (RQ), expressed by:

$$SQ = \frac{\sum_{(p,g)\in TP} IoU(pred, GT)}{|TP|} \tag{6.7}$$

$$RQ = \frac{TP}{|TP| + \frac{1}{2}\,|FP| + \frac{1}{2}|FN|} \tag{6.8}$$

6.3 Results

6.3.1 Metrics

The metrics section presents (3.1.1) semantic segmentation metrics, (3.1.2) instance segmentation metrics, and (3.1.3) panoptic segmentation metrics. The semantic segmentation metrics are related to the stuff classes in a per-pixel analysis. The instance segmentation classes relate to the thing classes using traditional object detection metrics, such as the AP. The panoptic segmentation metrics englobes both types of features.

Semantic Segmentation Results

Table 6.3 lists the general metrics for the three stuff categories (street, permeable area, and lake), considering the mIoU, fwIoU, mAcc, and pAcc for the Panoptic-FPN model with the ResNet-50 and ResNet-101 backbones. The validation and test results were very similar, in which the R101 backbone presented slightly better results among all metrics. In the validation and test sets, the metric with the most considerable difference between the ResNet-50 and ResNet-101 backbones was the IoU (0.514 and 1.484 difference in the validation and test set, respectively).

Table 6.4 lists the accuracy results of each "stuff" class for the validation and test sets. In addition to the three stuff classes (lake, permeable area, and street), the analysis creates another class merging the "thing" classes (we defined it as "all things"). Some samples have a single-class predominance, such as lake and permeable area, increasing the accuracy metric due to the high proportion of correctly classified pixels. The "lake" class presented the highest IoU for the validation (97.1%) and test (97.8%) sets, mainly because it presents very distinct characteristics from all other classes in the dataset. The permeable area achieves a slightly lower accuracy (IoU of 95.384 for validation and 96.275 for the test) than the lake class because it encompasses many different intraclass features (e.g., trees, grass, earth, sand). The "street" class, widely studied in remote sensing, presented an IoU of 88% and 90% for validation and testing. These IoU values are significant considering the difficulty of street mapping even by visual interpretation due to the high interference of overlapping objects (e.g., cars, permeable areas, undefined elements) and the challenges with shaded areas.

Table (6.3) Mean Intersection over Union (mIoU), frequency weighted (fwIoU), mean accuracy (mAcc), and pixel accuracy (pAcc) results for semantic segmentation in the BSB Aerial Dataset validation and test sets.

Backbone	**mIoU**	**fwIoU**	**mAcc**	**pAcc**
		Validation set		
R50	92.129	92.865	95.643	96.271
R101	92.643	93.241	95.769	96.485
Difference	0.514	0.376	0.126	0.214
		Test set		
R50	92.381	93.404	95.772	96.573
R101	93.865	94.472	96.339	97.148
Difference	1.484	1.068	0.567	0.575

Table (6.4) Segmentation metrics (Intersection over Union (IoU) and Accuracy (Acc)) for each stuff class in the BSB Aerial dataset validation and test sets considering the ResNet101 (R101), ResNet50 (R50) backbones, and their difference (R101-R50).

Category	R101		R50		Difference	
	IoU	**Acc**	**IoU**	**Acc**	**IoU**	**Acc**
			Validation set			
All things	89.962	95.060	89.402	94.882	0.56	0.178
Street	88.079	91.773	86.933	91.799	1.146	-0.026
Perm. Area	95.384	98.090	95.286	97.786	0.098	0.304
Lake	97.148	98.153	96.993	98.105	0.155	0.048
			Test set			
All things	90.718	94.563	89.142	93.041	1.576	1.522
Street	90.607	93.600	89.129	93.844	1.478	-0.244
P. Area	96.275	98.775	95.559	98.120	0.716	0.655
Lake	97.859	98.459	95.665	98.013	2.194	0.446

The R101 backbone presented better IoU results for all categories. The most significant

difference was the street category in the validation set (1.146) and the lake in the test set (2.194). The R50 backbone presented a higher value for the street class in the validation (0.026) and test sets (0.244). Since the balancing of the classes is not even, the IoU provides more insightful results when compared to the accuracy.

Instance Segmentation Results

Table 6.5 lists the results for the standard COCO metrics (AP, AP_{50}, AP_{75}, AP_S, AP_M, and AP_L) for the thing classes, considering the bounding box (Box) and segmentation mask (mask), from the two backbones (ResNet-101 (R101) and ResNet-50 (R50)). The validation and test results were very similar to those occurring in the stuff classes. However, the primary metric (AP) differences among the two backbones (R101 R50) were more considerable in the test set regarding the box metrics, with a difference of nearly 1.6%. The R101 backbone had higher values in almost all derived metrics, except for the AP75 box metric in the validation set and the APmedium in the test set.

Although the overall metrics showed better performance for the R101 backbone, the analysis by class presents some classes with slightly better results for the R50 backbone (Table 6.6). In the validation set, five of the eleven classes had higher values in the ResNet-50 backbone (harbor, boat, soccer field, house, and small construction). This effect was less frequent in the test set, showing only the boat class with superiority of the ResNet-50 backbone in the box metric and three classes (swimming pool, boat, and commercial building) in the mask metric.

Table (6.5) COCO metrics for the thing categories in the BSB Aerial Dataset validation set considering two backbones (ResNet-101 (R101) and ResNet-50 (R50)) and their difference (R101 R50).

Back-bone	**Type**	**AP**	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Validation set							
R101	Box	47.266	69.351	50.206	26.154	51.667	55.680
	Mask	45.379	68.331	50.917	24.064	49.490	57.882
R50	Box	45.855	68.258	51.351	25.806	49.732	48.678
	Mask	42.850	68.553	48.863	21.213	47.686	47.040
Difference	Box	1.411	1.093	-1.145	0.348	1.935	6.993
	Mask	2.529	2.778	2.054	2.851	1.804	10.842
Test set							
R101	Box	47.691	67.096	52.552	28.920	49.795	57.446
	Mask	44.211	65.271	49.394	25.016	49.377	58.311
R50	Box	44.642	64.306	50.727	28.636	49.881	53.298
	Mask	41.933	62.821	47.640	23.631	50.027	52.204
Difference	Box	3.049	2.790	1.825	0.284	-0.086	4.148
	Mask	2.278	2.450	1.754	1.385	-0.650	6.107

Table (6.6) AP metrics for bounding box and mask per category considering the thing classes in the BSB Aerial Dataset validation set for the ResNet101 (R101) and ResNet50 (R50) backbones and their difference (R101-R50).

Category	R101		R50		Difference	
	Box AP	Mask AP	Box AP	Mask AP	Box AP	Mask AP
Validation set						
Swimming pool	55.495	53.857	53.121	51.974	2.374	1.883
Harbor	37.137	21.079	39.415	24.300	-2.278	-3.221
Vehicle	55.616	56.573	54.568	55.893	1.048	0.680
Boat	30.582	36.216	35.329	37.265	-4.747	-1.049
Sports court	56.681	55.193	46.906	42.494	9.775	12.699
Soccer field	34.866	39.569	39.619	41.767	-4.753	-2.198
Com. building	32.114	31.799	28.592	28.471	3.522	3.328
Com. building block	66.283	63.192	52.149	47.606	14.134	15.586
Residential building	67.046	57.615	63.512	54.312	3.534	3.303
House	57.555	56.697	59.907	57.470	-2.352	-0.773
Small construction	26.550	27.381	31.284	29.800	-4.734	-2.419
Test set						
Swimming pool	53.561	50.044	51.546	50.520	2.015	-0.476
Harbor	42.429	22.837	31.409	17.270	11.02	5.567
Vehicle	56.371	57.689	55.695	57.311	0.676	0.378
Boat	26.190	31.210	30.698	34.875	-4.508	-3.665
Sports court	46.018	45.515	40.566	40.672	5.452	4.843
Soccer field	46.279	45.831	36.832	33.886	9.447	11.945
Com. building	42.516	37.709	41.145	40.265	1.371	-2.556
Com. building block	70.971	67.465	69.341	63.679	1.63	3.786
Residential building	54.829	47.397	51.774	44.640	3.055	2.757
House	62.395	59.886	57.861	58.396	4.534	1.490
Small construction	26.046	20.740	24.202	19.746	1.844	0.994

Panoptic Segmentation Results

Table 6.7 lists the results for the panoptic segmentation metrics (PQ, SQ, and RQ), which are the main metrics for evaluating this task. In hand with the previous stuff and thing results, the ResNet-101 backbone presented the best metrics in most cases, except for the RQ_{stuff} in the validation set and the SQ_{things} in the test set. Overall, the main metric for analysis (PQ) had nearly a 2% difference among the backbones. The low discrepancies among the different architectures suggest that in situations with lower computational power, the usage of a lighter backbone still presents close enough results.

6.3.2 Visual Results

Figure 6.8 shows five test and validation samples, including the original images and predictions from the Panoptic-FPN model using the ResNet-101 backbone. The results demonstrate a coherent urban landscape segmentation, visually integrating countable objects (things) and amorphous regions (things) in an enriching perspective toward real-world representation. Among

Table (6.7) COCO metrics for panoptic segmentation in the BSB Aerial Dataset validation and test sets considering the Panoptic Quality (PQ), Segmentation Quality (SQ), and Recognition Quality (RQ).

Backbone	Type	PQ	SQ	RQ
Validation Set				
R101	All	65.296	85.104	76.229
	Things	59.783	82.876	71.948
	Stuff	85.508	93.272	91.925
R50	All	63.829	84.886	74.550
	Things	57.958	82.777	69.674
	Stuff	85.354	92.617	92.432
Difference	All	1.467	0.218	1.679
	Things	1.825	0.099	2.274
	Stuff	0.154	0.655	-0.507
Test Set				
R101	All	64.979	85.378	75.474
	Things	58.354	83.171	69.997
	Stuff	89.272	93.468	95.558
R50	All	62.230	85.315	72.179
	Things	55.239	83.344	65.956
	Stuff	87.864	92.540	94.998
Difference	All	2.749	0.063	3.295
	Things	3.115	-0.173	4.041
	Stuff	1.408	0.928	0.560

the ten image pairs, there is at least one representation of each of the fourteen classes. As shown in the metrics section, the results present no evident discrepancies in the validation and testing data, demonstrating very similar visual results in both sets. The segmented images show the high ability to visually separate the different instances, even in crowded situations like cars in parking lots. Furthermore, the stuff classes are very well delineated, showing little confusion among the street, permeable areas, and lake classes. The set of established classes allows a good representation of the urban landscape elements, even considering some class simplifications. Therefore, panoptic segmentation congregates multiple competencies in computer vision for satellite imagery interpretation in a single structure.

6.4 Discussion

The panoptic segmentation task imposes new challenges in the formulation of algorithms and database structures, covering particularities of both object detection and semantic segmentation. Therefore, panoptic segmentation establishes a unified image segmentation approach, which changes digital image processing and requires new annotation tools and extensive and adapted datasets. In this context, this research innovates by developing a panoptic data annotation tool, establishing a panoptic remote sensing dataset, and being one of the first evaluations of the use of panoptic segmentation in urban aerial images.

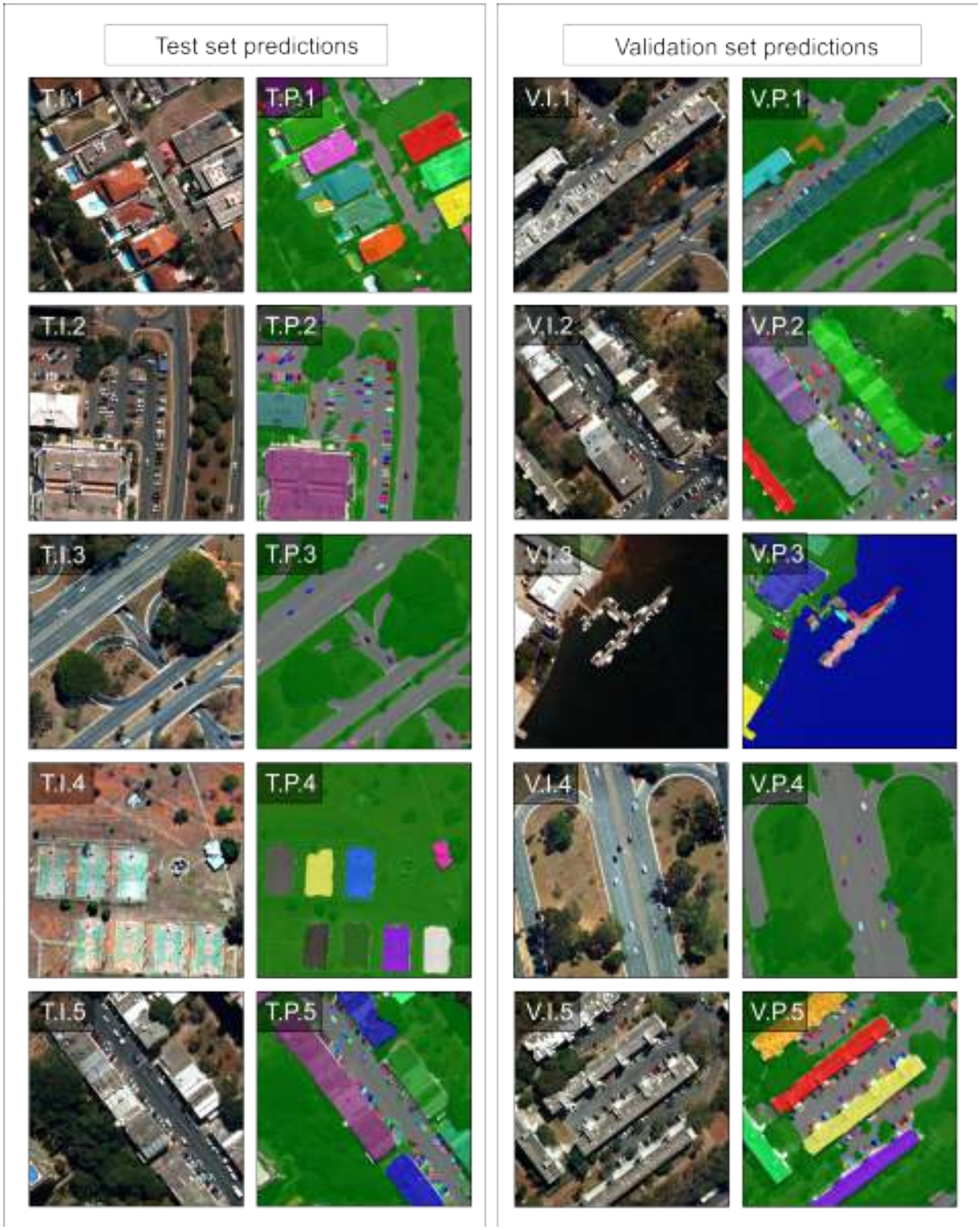

Figure (6.8) Five pair examples of validation images (V.I.1-5) and test images (T.I.1-5) with their corresponding panoptic predictions (V.P.1-5 and T.P.1-5).

6.4.1 Annotation Tools for Remote Sensing

Many software annotation tools are available online, e.g., LabelMe (Russell et al., 2008). Nevertheless, those tools have problems with satellite image data because of large sizes and other

singularities that are uncommon in the traditional computer vision tasks: (a) image format (i.e., satellite imagery is often in GeoTIFF, whereas traditional computer vision uses PNG or JPEG images), (b) georeferencing, and (c) compatibility with polygon GIS data. The remote sensing field made use of GIS software long before the rise of DL. With that said, there are extensive collections of GIS data (urban, agriculture, change detection) that other researchers could apply DL models. However, vector-based GIS data requires modifications to use DL models. We proposed a conversion tool from GIS data that automatically crops image tiles with their corresponding polygon vector data stored in shapefile format to panoptic, instance, and semantic annotations. The proposed tool is open access and works independently, without the need to use proprietary programs such as LabelRS developed by ArcPy and dependent on ArcGis (Li et al., 2021). Our proposed pipeline and software enable the users to choose many samples for training, validation, and testing in strategic areas using point shapefiles. This method of choosing samples presents a huge benefit compared to methods such as sliding windows for image generation. Finally, our software enables the generation of the three segmentation tasks (instance, semantic, and panoptic), allowing other researchers to exploit the field of desire.

6.4.2 Datasets

Most transfer learning applications use trained models from extensive databases such as the COCO dataset. Nevertheless, remote sensing images present characteristics that may not yield the most optimal results using traditional images. These images contain diverse targets and landscapes with different geometric shapes, patterns, and textural attributes, representing a challenge for automatic interpretation. Therefore, the effectiveness of training and testing depends on accurately annotated ground truth datasets, which requires much effort to build large remote sensing databases with a significant variety of classes. Furthermore, the availability of open access encourages new methods and applications, as seen in other computer vision tasks.

Lin et al. (2019) performed a complete review of remote sensing image datasets for DL methods, including tasks of scene classification, object detection, semantic segmentation, and change detection. In this recent review, there is no panoptic segmentation database, demonstrating a knowledge gap. Most datasets consider limited semantic categories or target a specific element, such as building (Benedek et al., 2012; Ji et al., 2019b; Van Etten et al., 2018), vehicle (Drouyer, 2020; Lin et al., 2020; Zeng et al., 2021), ship (Hou et al., 2020; Huang et al., 2018; Wei et al., 2020), road (Das et al., 2011; Maggiori et al., 2017), among others. Regarding available remote sensing datasets for various urban categories, one of the main is the iSAID (Waqas Zamir et al., 2019), with 2,806 aerial images distributed in 15 different classes, for instance segmentation and object detection tasks.

The scarcity of remote sensing databases with all cityscape elements makes mapping difficult due to highly complex classes, numerous instances, and mainly intraclass and interclass elements commonly neglected. Adopting the panoptic approach allows us to relate the content of interest and the surrounding environment, which is still little explored. Therefore, organizing large

datasets into panoptic categories is a key alternative to mapping complex environments such as urban systems that are not reached even with enriched semantic categories.

The proposed BSB Aerial Dataset contains 3,400 images (3,000 for training, 200 for validation, and 200 for testing) with 512x512 dimensions containing fourteen common urban classes. This dataset simplified some urban classes, such as sports courts instead of tennis courts, soccer fields, and basketball courts. Our dataset considers three stuff classes, widely represented in the urban setting, such as roads. The availability of data and the need for periodic mapping of urban infrastructure by the government allows for the constant improvement of this database.

6.4.3 Difficulties in the Urban Setting

Although this study shows a promising field in remote sensing with a good capability of identifying thing and stuff categories simultaneously, we observed four main difficulties in image annotation and possible results in the urban setting (Figure 6.9): (1) shadows, (2) occlusion objects, (3) class categorization, and (4) edge problem on the image tiles. Shadows entirely or partially obstruct the light and occur under diverse conditions from the different objects (e.g., clouds, buildings, mountains, and trees), requiring well-established ground rules to obtain consistent annotations. Therefore, the shadow presence is a source of confusion and misclassification, reducing image quality for visual interpretation and segmentation and, consequently, negatively impacting the accuracy metrics (Wang et al., 2017) (Figure 6.9A1, 6.9A2, and 6.9A3). Specifically, urban landscapes have a high proportion of areas covered by shadows due to the high density of tall objects. Therefore, urban zones aggravate the interference of shadows, causing semantic ambiguity and incorrect labeling, which is a challenge in remote sensing studies (Lin et al., 2019; Liu et al., 2018b). DL methods tend to minimize shading effects, but errors occur in very low-light locations. Another fundamental problem in computer vision is the occlusion that impedes object recognition in satellite images. Commonly, there are many object occlusions in the urban landscape, such as vehicles partially covered by trees and buildings, making their identification difficult even for humans (Figure 6.9B1, 6.9B2, and 6.9B3).

Like the occlusion problem, the objects that rely on the tile edges may present an insufficient representation. In monothematic studies, the authors may design the dataset to avoid this problem. However, for the panoptic segmentation task, which aims for an entire scene pixel-wise classification, some objects will be partial representation no matter how large we choose the image tile (Figure 6.9D1, 6.9D2, and 6.9D3). Our proposed annotation tool enables the authors to select each tile's exact point, which gives data generation autonomy to avoid very few representations (even though the problem will still be present). By choosing large image tiles, the percentual representation of edge objects will be lower and tends to have a smaller impact on the model and accuracy metrics but increasing the image tile also requires more computational power.

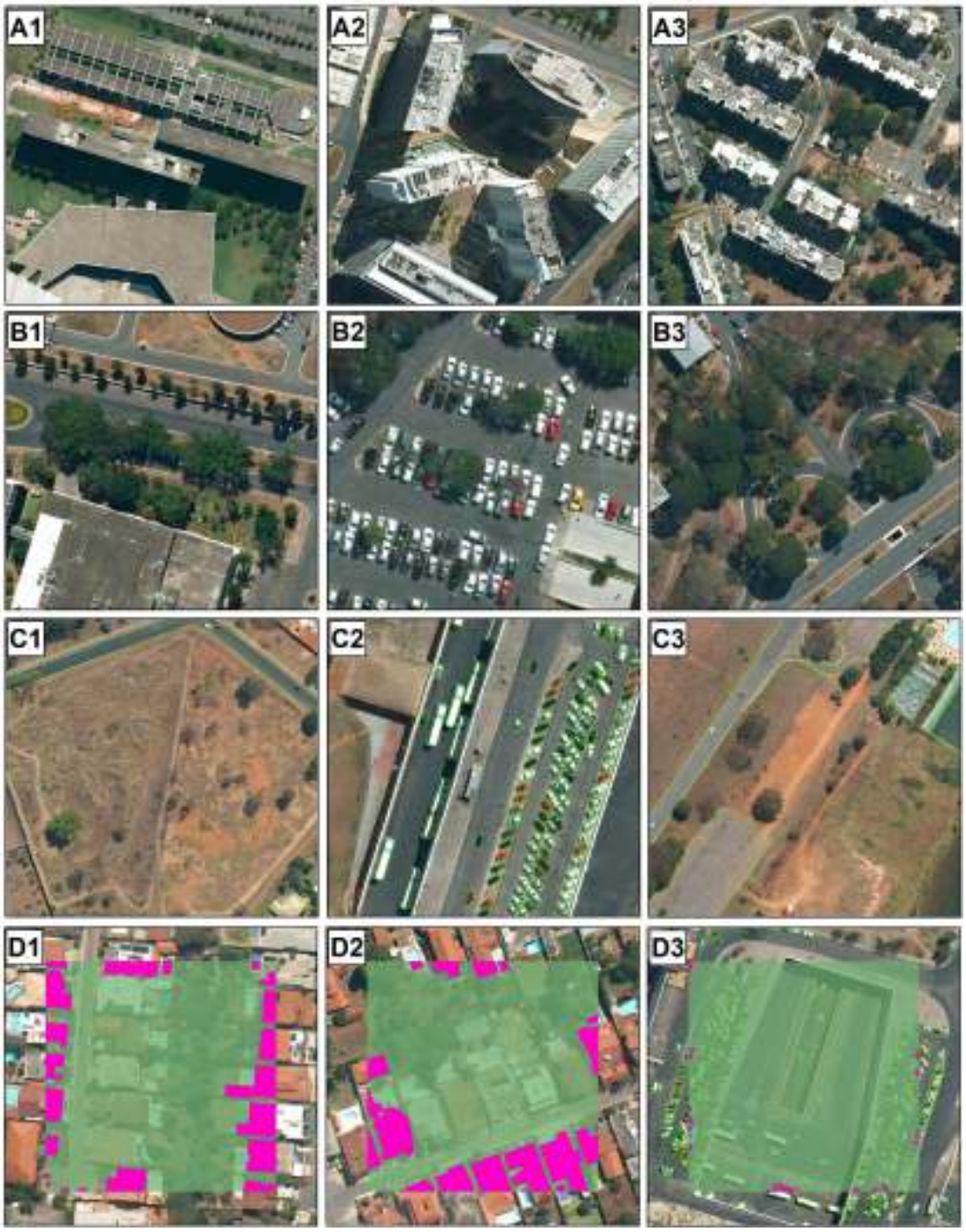

Figure (6.9) Three examples of (1) shadow areas (A1, A2, and A3), (2) occluded objects (B1, B2, B3), (3) class categorization (C1, C2, and C3), and (4) edge problem on the image tiles (D1, D2, and D3).

Finally, the improvement of urban classes in the database is ongoing work. This research sought to establish general and representative classes, but the advent of new categories will allow for more detailed analysis according to research interests. For example, our vehicle class encompasses buses, small cars, and trucks, and our permeable area class contains bare ground, grass, and trees as shown in Figures 6.9C1, 6.9C2, and 6.9C3.

6.4.4 Panoptic Segmentation Task

The remote sensing field is prone to using panoptic segmentation, mainly when referring to satellite and aerial images that do not require real-time processing. Most images have a frequency of at least days apart from each other, making some widely studied metrics such as inference time much less relevant. In remote sensing, the more information we can get simultaneously, the better. However, panoptic segmentation presents some non-trivial data generation mechanisms that require information for both instance and semantic segmentation. The existing panoptic segmentation studies that develop novel remote sensing datasets do not fully embrace the "stuff" classes (Garnot and Landrieu, 2021; Khoshboresh-Masouleh and Shah-Hosseini, 2021).

The panoptic segmentation may represent a breakthrough in the remote sensing field for the ability to gather countable objects and background elements using a single framework, surpassing some difficulties of semantic and instance segmentation. Nonetheless, the models' data generation process and configuration are much less straightforward than other methods, highlighting the importance of shortening this gap.

6.4.5 Limitations and Future Work

The high diversity of properties in remote sensing images (different spatial, spectral, and temporal resolutions) and the different landscapes of the Earth's surface make it challenging to formulate a generalized DL dataset. In this sense, our proposed annotation tool is suitable for creating datasets considering different image types. Future research on panoptic segmentation in remote sensing should progress to include images from various sensors, allowing faster advances in its application.

Furthermore, an important advance for panoptic segmentation is to include occlusion scenarios. Currently, the panoptic segmentation and its subsequent metrics (PQ, SQ, and RQ) require no overlapping segments, i.e., it considers only the visible pixels of the images. The usage of top-view images is very susceptible to classifying non-visible areas (occluded targets). Those changes would require adaptations in the models and metrics.

Practical remote sensing applications also require mechanisms for classifying large regions. Those methods usually use sliding windows, which have different peculiarities for pixel-based (e.g., semantic segmentation) and box-based methods (e.g., instance segmentation). The semantic segmentation approach use sliding windows with overlapping pixels, in which overlapped pixels are averaged. This averaging procedure attenuates the borders and enhances the metrics (Costa et al., 2021; da Costa et al., 2021a; de Albuquerque et al., 2020a). The instance segmentation proposals use sliding windows with a half-frame stride value, which allows identifying the elements as a whole and eliminating partial predictions (Carvalho et al., 2021; de Carvalho et al., 2021c). There is no specific method for using a panoptic segmentation framework using sliding windows.

6.5 Conclusions

The application of panoptic, instance, and semantic segmentation often depends on the desired outcome of a research or industry application. Nevertheless, a research gap in the remote sensing community is the lack of studies addressing panoptic segmentation, one of the most powerful techniques. The present research proposed an effective solution for using this unexplored and powerful method in remote sensing by: (a) providing a large dataset (BSB aerial dataset) containing 3,400 images with 512x512 pixel dimensions in the COCO annotation format and fourteen classes (eleven "thing" and three "stuff" categories), being suitable for testing new DL models, (b) providing a novel pipeline and software for easily generating panoptic segmentation datasets in a format that is compatible with state-of-the-art software (e.g., Detectron2), and (c) leveraging and modifying structures in the DL models for remote sensing applicability, and (d) making a complete analysis of different metrics and evaluating difficulties of this task in the urban setting. One of the main challenges for preparing a panoptic segmentation model is the image format, which is still not well documented. We proposed an automatic converter from GIS data to panoptic, instance, and semantic segmentation formats. GIS data was widespread even before the DL rise, and the number of datasets that could benefit from our method is enormous. Besides, our tool allows the users to choose the exact points in large images to generate the DL samples using point shapefiles, which brings more autonomy to the studies and allows better data choosing. We believe this work may increase other studies on the panoptic segmentation task with the BSB Aerial Dataset, the annotation tool, and the baseline comparisons using well-documented software (Detectron2). Moreover, we evaluated the Panoptic-FPN model using two backbones (ResNet-101 and ResNet-50), showing promising metrics for this method's usage in the urban setting. Therefore, this research shows an effective annotation tool, a large dataset for multiple tasks, and their application on some non-trivial models. Regarding future studies, we discussed three major problems to be addressed: (1) augmenting the dataset with images with different spectral bands and spatial resolution, (2) expanding the panoptic idea for occlusion scenarios in remote sensing, and (3) adapting methods for classifying large images.

Chapter 7

Rethinking Panoptic Segmentation

This chapter uses the previously developed dataset, bringing a novel and simpler methodology to obtain panoptic predictions while still using semantic segmentation models. Basically, this chapter removes the complexity of box-based methods using learning mechanisms and exchanges by non-learning simple image processing steps. The results from this chapter were submitted to IEEE Geoscience and Remote Sensing Letters, and it is currently past the first round of review.

7.1 Presentation

As shown in the previous chapter, the panoptic segmentation task is very little exploited in the remote sensing community. The previous chapter addressed a methodology for adjusting GIS data to the commonly used COCO panoptic annotation format. The fewer number of panoptic segmentation papers can also be explained by additional difficulties in the data preparation, model configuration, and necessary post-processing applications for remote sensing, explained as follows.

First, the panoptic segmentation models require the data in a specific structured format. For example, Detectron2 (Wu et al., 2019) (one of the most used open software for instance and panoptic segmentation) needs the data in the Common Objects in Context (COCO) annotation format (Lin et al., 2014). In this context, de Carvalho et al. (2021b) pointed out this issue and proposed a conversion software from GIS raster data to panoptic data in the COCO format. However, the proposed pipeline presents many steps. In contrast, the semantic segmentation data only demands a ground truth image, requiring the conversion from polygon shapefile to raster data in Geographic Information System (GIS) software.

Second, the box-based panoptic segmentation models (e.g., Panoptic-FPN (Kirillov et al., 2019), EfficientPS (Mohan and Valada, 2021b)) use an instance segmentation module (usually the Mask-RCNN He et al. (2017)), introducing a complexity to the entire process. For example, the loss function for the instance segmentation module includes box regression, mask loss, and classification loss, and there is a higher number of hyperparameters (e.g., anchor box sizes, aspect ratios). The semantic segmentation models usually use a single loss function, and few hyperparameters are necessary. Therefore, some studies target instance segmentation from se-

mantic segmentation models, considering a border insertion approach to identify unique objects (de Carvalho et al., 2021a; Heidler et al., 2021; Mou and Zhu, 2018). However, these studies considered a single class, and leverages are necessary to expand this approach to multiclass panoptic segmentation.

Finally, large image classification using deep learning requires a sliding window approach. To the best of our knowledge, there are still no attempts to extend panoptic segmentation models for large scene classification. The semantic segmentation method generally considers the classification of consecutive frames, where a stride smaller than the frame dimension results in overlapping pixels, the average of which results in smoothed edges and better accuracy metrics (de Albuquerque et al., 2020a). The instance segmentation approach is a little more complicated, in which the authors use a half-frame stride value and non-maximum suppression to maintain only the largest bounding boxes, which represents a total prediction (Carvalho et al., 2021).

This study proposes an interpretation of panoptic segmentation as an extension of the semantic segmentation task with a few post-processing steps that do not require learning methods for unique object segmentation. We replace the instance segmentation module for a simple change in the data preparation (apply borders on polygons that may merge) and a few post-processing steps (isolate object interiors, attribute unique values, and expand objects with deleted borders). This approach reduces the number of steps in the data generation process, the models do not need instance segmentation modules, and it enables the implementation of sliding windows straightforwardly.

7.2 Materials and Methods

7.2.1 Dataset

The BSB Aerial Dataset (de Carvalho et al., 2021b) is a publicly open panoptic dataset containing 3,400 image tiles and polygon shapefiles compatible with Geographic Information Systems (GIS) software. The image data includes the Red, Green, and Blue (RGB) channels obtained by an aerial flight over the city of Brasília, Brazil. The spatial resolution is 24 centimeters, sufficient to observe features such as cars. The shapefile contains fourteen classes, three "stuff" (road, permeable area, lake), and eleven "thing" categories (swimming pool, harbor, ground vehicle, water vehicle, sports court, soccer field, commercial building, commercial building block, house, small construction).

7.2.2 Preparation Pipeline Using GIS software

The BSB Aerial Dataset (de Carvalho et al., 2021b) obtained the panoptic annotations from the development of software for converting GIS data to the COCO annotation format, requiring a semantic segmentation and a sequential mask (that is, each polygon belonging to the "things" category has a single value). In contrast, the present proposition only requires a single mask without needing a conversion system, containing all elements belonging to the "things" category individually separated.

However, semantic segmentation models present problems in uniquely identifying objects of the same class that are in contact since the prediction will merge many objects in a single polygon. One solution is to insert borders in objects to isolate their interiors and generate individual polygons. Some classes are prone to merge (e.g., cars and houses, see Fig 7.1C and 7.1D), while others will never have this problem (e.g., harbor and residential buildings, see Fig 7.1A and 7.1B).

Figure (7.1) Examples of non-merging categories (A and B), and merging categories (C and D).

We subdivided the "things" categories into (1) merging classes and (2) non-merging classes. We applied a negative buffer for all merging "thing" classes, creating a 1-pixel border inside the polygon features and avoiding overlapping. Border areas correspond to their own classes (i.e., vehicle contour, boat contour, house contour), making it easy to apply weights to each class based on its representation. Seven of the eleven thing classes had merging possibilities (vehicle, boat, sports court, commercial building, commercial building block, house, and small construction), resulting in 22 classes. The dataset considered 3,000, 200, and 200 image tiles (512x512 pixel dimensions) for training, validation, and testing.

7.2.3 Deep Learning Model

Any semantic segmentation model can be adapted to obtain panoptic predictions. This study used the U-net architecture (Ronneberger et al., 2015a) with the Efficient-net backbone (Tan and Le, 2019a). The U-net is an encoder-decoder structure in which the encoder extracts features and the decoder restores the image dimensions for precise pixel classification. The Efficient-net

backbone is a convolutional neural network that uses the width, depth, and resolution scaling, presenting eight configurations from B0 (less complex, i.e., fewer parameters) to B7 (more complex). The Efficient-net-B5 is the most complex configuration that runs in our graphics processing unit (NVIDIA RTX with 11GB RAM). We compared three levels of complexity: Efficient-net-B0, Efficient-net-B3, and Efficient-net-B5.

To ensure the comparison, the model training used the same: (a) hyperparameters (Adam optimizer, batch size of 4, 200 epochs, and learning rate of 0.0001), (b) loss function (weighted cross-entropy loss, in which the weights used the number of pixels of the most represented class divided by the number of pixels from the current class), and (c) augmentations (random horizontal and vertical flips with 50% probability). The definition of the best model considers the least loss of validation.

7.2.4 Semantic to Panoptic Segmentation Algorithm

Our model outputs a pixel-wise classification with semantic labels, but the borders are vital for separating each instance adequately (avoiding possible class merges). The proposed algorithm then receives five inputs: (1) the semantic segmentation prediction (with borders), (2) a list of stuff classes, (3) a list of thing classes, (4) a list of thing classes that have borders, (5) a list of minimum polygon area for each class, and (6) a list of priority classes. The process considers four steps for each class (`https://github.com/osmarluiz/semantic2panoptic`):

1. Create a list with the polygon coordinates using OpenCV find contours.
2. Rank order the polygons by their area and eliminate polygons according to a predefined threshold value adjusted per category to avoid noisy representations (e.g., a 10-pixel representation of houses is probably a noisy representation).
3. Transform each polygon (set of coordinates) to mask (filled pixels). This step can attribute distinct values to each object - in which each object receives a different number in ascending order. This procedure considers closed polygons with their outer contours and may present errors in overlapping object cases (e.g., swimming pool on top of a building), breaking the panoptic principle of non-overlapping classes. To solve this problem, we must ensure the order of priority, resulting in a single value for each pixel.
4. If the object is a thing class with borders, we apply the expanding border (EB) algorithm, which makes eight copies of the semantic prediction with a 1-pixel dislocation in all directions (up, down, left, right, up-right, up-left, down-right, and down-left) and apply the logical OR operation to all channels. This procedure results in one-pixel expansion for all objects with deleted borders maintaining the correct individual value for each one. Since the borders are inside each polygon, in principle, the expanding border algorithm will not result in overlap between close objects.

7.2.5 Sliding Window Approach

In semantic segmentation, using low stride values in the sliding windows generate overlapping pixels, which smoothens the frame borders and presents better accuracy metrics (de Albuquerque et al., 2020a). However, if we apply the overlapping pixels on the panoptic predictions, each pixel's values would change according to the number of overlaps, bringing wrong results. We must apply the sliding window on the semantic predictions (with borders), and the semantic to the panoptic algorithm is only applied in the final image. We reported the metrics on the sliding window with a stride value of 16 pixels (32 times smaller than the frame dimension).

7.2.6 Accuracy Analysis

The accuracy analysis evaluated three investigations: (1) ablation study to verify the EB algorithm varying the model backbone, (2) panoptic segmentation evaluation, and (3) sliding window evaluation. The performance of the EB algorithm used per-pixel metrics since it is just a fine adjustment of the polygons, and per-polygon metrics would be a little informative. The Intersection over Union (IoU) is adequate due to a low class-imbalance effect, being expressed by: $\frac{TP}{TP+FP+FN}$, where TP, FP, and FN represent true positives, false positives, and false negatives.

Then, with the best model, we used the traditional panoptic segmentation metrics proposed by Kirillov et al. (2019). The Panoptic Quality (PQ) is the multiplication of the Segmentation Quality (SQ): $SQ = \frac{\sum_{(p,g)\in TP} IoU(p,g)}{|TP|}$ with the Recognition quality (RQ): $RQ = \frac{TP}{|TP|+\frac{1}{2}|FP|+\frac{1}{2}|FN|}$, where p and g are the prediction and ground truth. It is important to note that this metric considers a TP as an object with an IoU above 0.5.

Finally, the evaluation of the sliding window approach considered the frequency of merged objects that are not represented by per-pixel metrics or by the panoptic metric. This analysis considered a new independent image of 2560x2560 pixels (25 times larger than the frames used for training) containing a parking lot (i.e., many situations for merging vehicles) and many houses. We evaluated TP, FP, FN, and merged polygons in this test area, where a TP is a polygon greater than 0.5 IoU.

7.3 Results and Discussion

7.3.1 Expanding Border Algorithm

Table 7.2 lists the IoU results with and without the EB algorithm for the three models. Even for elements that did not use EB, there is a small difference in the metrics due to the find countors procedure that may present a slight pixel dislocation. Those differences were consistently lower than 1%. Using more parameters increased the IoU and made the EB algorithm more effective, in which the difference between using the EB algorithm was 3.29% (70.01 against 66.72), 3.01% (69.72 against 66.71), and 2.18% (67.09 against 64.91) for the Efficient-net-B5, Efficient-net-B3, and Efficient-net-B0. Even though the border expansion increases the objects by only 1 pixel, the smaller objects are significantly affected. The vehicle class presents a difference of nearly

20% in the IoU score, showing the importance of the method for obtaining accurate predictions. For larger object classes, i.e., soccer field (SF), commercial building (CB), commercial building block (CBB), house, and small construction (SmC), the expanding border algorithm presented higher results, but with much less difference.

7.3.2 Semantic to Panoptic Segmentation

Table 7.3 lists the panoptic segmentation metrics (PQ, RQ, and SQ) for the best model (U-net with the Efficient-net-B5 backbone). The SQ presented metrics over 65% for all categories, and apart from the soccer field, all RQ metrics were above 50%. Since the PQ metric is the multiplication of SQ and RQ, five categories presented values below 50%: harbor (2), sports court (3), soccer field (2), commercial building(1), and small construction (1). When considering the mean average (mAvg), those classes with much lower values than the rest (e.g., soccer field) have a great impact, emphasizing the importance of evaluating them separately. Another supplementary metric that gives an overall perspective is the weighted average, which brings results nearly 20% higher since most classes that ate more recurring have less representation. Especially regarding soccer fields, the deep learning models may present difficulties since it is a kind of permeable area.

Figure 7.2 shows an example image from the test set (A) with its corresponding semantic segmentation ground truth (B), the prediction with borders (C), and the prediction after applying the proposed semantic to the panoptic algorithm (D), demonstrating the efficiency of separating objects using this method and providing accurate panoptic segmentation predictions. This approach can be regarded as an evolution to Mou and Zhu (2018), and de Carvalho et al. (2021a) methods, with the application for instance segmentation purposes.

7.3.3 Sliding Window Approach

Figure 7.3 shows the original RGB image (Fig. 7.3A), a zoom area (Fig. 7.3A1), and their respective panoptic predictions (Fig. 7.3B and Fig. 7.3B1). The sliding window method has an increasing computational cost when decreasing the stride value. Using a low stride value for extremely large areas may compromise the inference time. In our study, this independent test area with 2560x2560 pixel dimensions registered 3, 10, 38, 127, 435, 1837 seconds for strides of 512, 256, 128, 64, 32, and 16 pixels, respectively.

Table 7.1 lists the results for the sliding window approach for the 2560x2560-pixel test area. The image contained five classes (swimming pool, vehicle, sports court, house, and small constructions). The vehicles were the most prevalent class and presented 805 correct, 25 false positives, and 7 merged objects. Even though merging is something we want to avoid as much as possible, the ratio of merges by correct predictions is less than 1% for all classes.

The method proposed shows to be very efficient for mapping regions from remotely sensed images. This approach may limit other problems in the computer vision community that require real-time processing. In remote sensing, this issue is not much relevant since the frequency of images is at least days apart from each other. Semantic segmentation models may present noisy

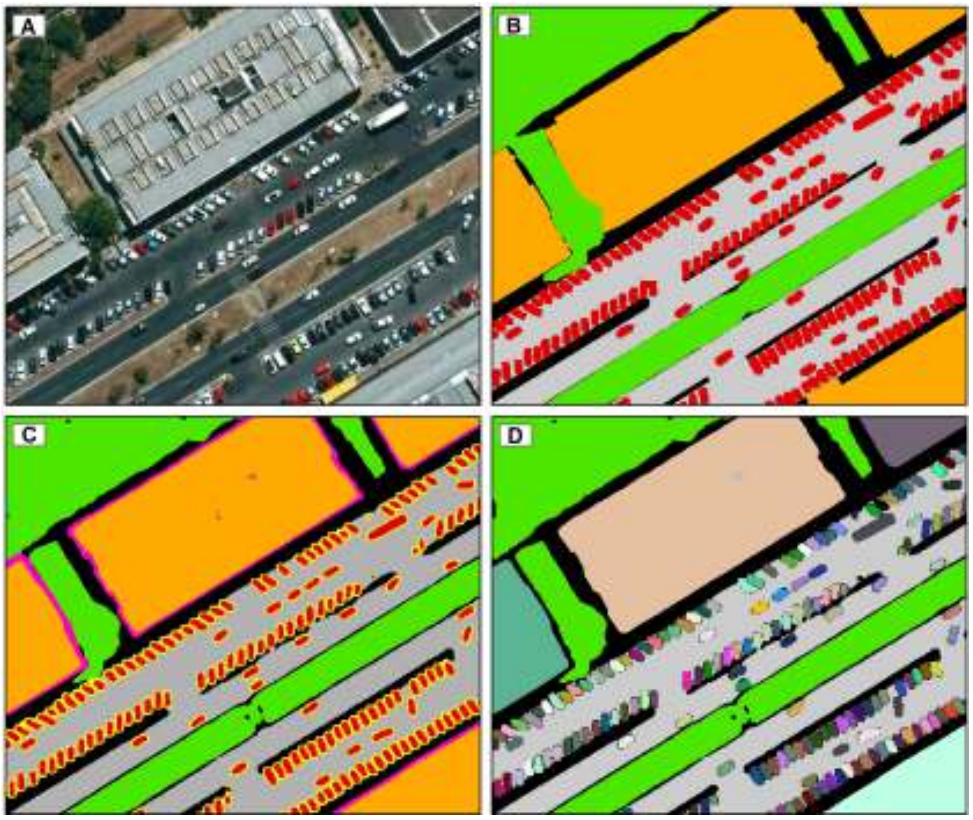

Figure (7.2) Example from the test set considering the: (A) original image, (B) semantic segmentation ground truth, (C) prediction with the borders, (D) panoptic prediction.

Table (7.1) Per object metrics and analysis on all classes for the 2560x2560 image considering: road, permeable area (PA), lake, swimming pool (SP), harbor, vehicle, boat, soccer field (SF), commercial building (CB), commercial building block (CBB), residential building (RB), house, and small construction (SmC).

	SP	vehicle	SC	house	SmC
TP	82	805	3	196	101
FP	2	25	1	6	11
FN	0	0	0	2	1
merged	0	7	0	3	0
accuracy	97.62	96.18	75.00	94.68	90.18

Figure (7.3) Original 2560x2560-pixel image (A) with its corresponding sliding windows panoptic results (A1) and a zoomed area from the image (A1) and prediction (B1).

Table (7.2) Intersection over Union (IoU) results for the three backbones, considering expanded border (EB) algorithm, and with no expanding borders (NB) for the fourteen classes: road, permeable area (PA), lake, swimming pool (SP), harbor, vehicle, boat, soccer field (SF), commercial building (CB), commercial building block (CBB), residential building (RB), house, and small construction (SmC). "*" denotes objects that presented borders.

	road	PA	lake	SP	harbor	vehicle*	boat*	SC*	SF	CB*	CBB*	RB	house*	SmC*	mIoU
Efficient-net-B5															
EB	83.56	91.86	97.28	67.06	53.61	83.75	51.69	67.60	52.20	66.82	70.75	84.32	78.18	31.38	70.01
NB	83.56	91.86	97.28	66.86	53.20	60.17	43.85	64.02	51.84	65.09	69.1	84.61	74.05	28.7	66.72
Dif	0	0	0	0.20	0.41	23.58	7.84	3.58	0.36	1.73	1.65	-0.29	4.13	2.68	3.29
Efficient-net-B3															
EB	82.37	91.07	97.38	64.78	55.77	81.27	42.82	62.40	71.48	63.39	72.83	82.93	77.58	30.03	69.72
NB	82.37	91.07	97.38	64.58	55.47	59.91	37.25	59.62	69.45	61.63	70.25	83.02	73.13	28.70	66.71
Dif	0	0	0	0.20	0.30	21.63	5.57	2.78	2.03	1.76	2.58	-0.09	4.45	1.33	3.01
Efficient-net-B0															
EB	82.00	89.66	96.44	63.74	42.81	77.65	32.65	65.17	68.11	63.72	72.37	81.08	75.79	28.08	67.09
NB	82.00	89.66	96.44	63.60	42.07	57.96	36.44	61.87	68.04	61.93	69.95	81.37	70.97	26.44	64.91
Dif	0	0	0	0.14	0.74	19.69	-3.79	3.30	0.07	1.79	2.42	-0.29	4.82	1.64	2.18

Table (7.3) Panoptic Quality (PQ), Segmentation Quality (SQ), and Recognition Quality (RQ) metrics for all classes: road, permeable area (PA), lake, swimming pool (SP), harbor, vehicle, boat, soccer field (SF), commercial building (CB), commercial building block (CBB), residential building (RB), house, and small construction (SmC) and their mean average (mAvg) and weighted average (wAvg). "*" denotes objects that presented borders.

	road	PA	lake	SP	harbor	vehicle*	boat*	SC*	SF	CB*	CBB*	RB	house*	SmC*	mAvg	wAvg
PQ	78.10	89.16	93.26	58.63	38.03	66.14	60.06	42.21	11.60	46.19	52.54	70.76	63.19	46.69	47.41	65.15
SQ	79.47	89.85	95.06	75.04	65.99	68.70	75.95	80.99	92.81	83.75	90.40	83.30	82.60	74.06	81.28	71.49
RQ	98.28	99.23	98.11	78.14	57.63	96.27	79.07	52.12	12.50	55.16	58.12	84.95	76.50	63.04	58.33	91.13

features, which are very small polygons. To solve this issue, we proposed a threshold to delete the contours with an area smaller than a specific value, which can change for each class. Future studies may benefit from applying learning mechanisms to obtain optimal threshold values for each class. Even though the number of merged polygons is very low (shown by the ratio of merges by true positive predictions), a possible solution for future studies is to make more specific classes, e.g., separate vehicles into cars, buses, and trucks. In this way, the average size of each target can help in the evaluation. For example, a car has an average dimension of 20x10 pixels, so if the prediction is double, it assumes the presence of two objects.

7.4 Conclusion

The present letter proposed an efficient algorithm to achieve panoptic predictions using semantic segmentation models, achieving good per-pixel results and a good capability of separating different instances. The data preparation is straightforward and can be quickly obtained in GIS software by generating borders on the polygons (using a negative buffer to avoid overlapping classes). The method introduced a non-learning method for separating different instances, which reduces the total number of parameters. We show that any semantic segmentation model applies; however, better models may yield better results for our expanding border algorithm. Besides, great importance for remote sensing studies involves the classification of large areas, and there are no works regarding this topic for panoptic segmentation. Our approach makes it very easy to apply sliding windows since it uses the same semantic segmentation logic with the semantic to panoptic algorithm as the last step.

Chapter 8

Multispectral Panoptic Segmentation

This chapter brings the first study considering the panoptic segmentation task in the beach landscape and also using multispectral data. The results were submitted and accepted by the IEEE International Geoscience and Remote Sensing Symposium, and expanded results were published in the International Journal of Applied Earth Observations and Geo-Information.

8.1 Presentation

Many studies have been performed on panoptic segmentation for ground-level RGB images (Cheng et al., 2020a; Xiong et al., 2019), medical images (Cha et al., 2021; Yu et al., 2020; Zhang et al., 2018a), and videos (Kim et al., 2020; Qiao et al., 2021). Panoptic segmentation was first explored in orbital or aerial remote sensing data (top-view images) only recently, presenting few studies (de Carvalho et al., 2022c; De Carvalho et al., 2022; Garnot and Landrieu, 2021; Hua et al., 2021; Khoshboresh-Masouleh and Shah-Hosseini, 2021). Among those studies, Garnot and Landrieu (2021) used remote sensing peculiarities in terms of spectral bands. However, the authors considered a dataset containing only "thing" classes. Khoshboresh-Masouleh and Shah-Hosseini (2021) evaluated the change detection in very high-resolution Google Earth images but only considered the building class (things). de Carvalho et al. (2022c) developed a dataset with "thing" and "stuff" classes, but they used an aerial image only containing the RGB channels, being very similar to traditional ground-level images. Finally, Hua et al. (2021) used datasets previously designed for instance segmentation but adapted for the panoptic segmentation task, considering only RGB images.

Therefore, the multispectral imaging dataset has not yet been explored in panoptic segmentation. Satellite or aircraft-based images often present many different characteristics, such as large spatial dimensions, varying number of channels, image format, and georeferencing (Carvalho et al., 2021). Furthermore, in the field of remote sensing technologies for monitoring the Earth's surface, there is a wide variety of images (multispectral, hyperspectral, Synthetic Aperture Radar (SAR), and thermal) coming from different platforms (satellites, Unmanned

Aerial Vehicles (UAV), and aerial images). The particularities of orbital and aerial images differ from datasets produced by the computer vision community such as Common Objects in Context (COCO) (Lin et al., 2014), Mapillary Vistas (Neuhold et al., 2017), Cityscapes (Cordts et al., 2016), which contains Red, Green, and Blue (RGB) images at a ground level. The development of new tasks, such for instance segmentation (He et al., 2020), panoptic segmentation (Kirillov et al., 2019) and eventual novel methods (Bolya et al., 2020; Gao et al., 2021; Mohan and Valada, 2021a) are all designed in the first moment for those traditional ground-level RGB images. Therefore the orbital and aerial image peculiarities require specific software and methodologies to extract the most out of it since even preliminary stages such as generating image tiles with a specific size and annotation format may be challenging (de Carvalho et al., 2022c; Li et al., 2021). Most software that is openly available today, such as Facebook's Detectron2 (Wu et al., 2019) is designed with specifications for RGB images with three channels in conventional formats such as Joint Photographics Experts Group (JPEG) and Portable Network Graphics (PNG). Adapting those configurations may not be straightforward, making using some new methods much harder in satellite or aircraft-based remote sensing.

The continuous monitoring and inspection of tourist activity along the beaches is essential for achieving effective public and environmental policies. In this context, panoptic segmentation from the remote sensing images can facilitate the inspection process. However, few beach studies used remote sensing data with deep learning. The beach scene is mainly composed of small objects, which are represented by few pixels even with high-resolution images, being a significant challenge. Deep learning models generally perform poorly on small targets due to their noisy representation and confusion with other targets (de Carvalho et al., 2021d; Tong et al., 2020).

This study aims to introduce panoptic segmentation with multispectral remote sensing data, providing theory, application, and methods contributions as follows:

1. We aim to verify the importance of band selection within the panoptic segmentation task and compare the conceptual results of panoptic, instance, and semantic segmentation.

2. A viable and state-of-the-art application for beach inspection, being the first study to explore panoptic segmentation in the beach setting, providing a novel dataset comprising thirteen classes and benchmark results for panoptic and semantic segmentation.

3. The panoptic segmentation task was initially developed for RGB images, and the present research carried out changes in the original code to allow the joint processing of a varying number of spectral bands. Besides, this study adjusted the ResNeXt-101 backbone for Panoptic-FPN.

8.2 Material and methods

8.2.1 Study area

The study area is located in the Praia do Futuro region, Fortaleza, Brazil, with intense tourist and economic activity. Figure 8.1A shows the study area highlighted in yellow borders, and Figure 8.1B shows a zoomed area containing tourist umbrellas, beach umbrellas, suns, straw sun, trees, buildings, and swimming pools.

Figure (8.1) Study Area, in which (A) shows the region in Brazil, (B) shows a more detailed zoom of the beach area considered in this study, and (C) shows a larger zoom to show what kind of elements is visible with the WorldView-3 images.

8.2.2 Image acquisition and annotations

We used WorldView-3 images provided by the European Space Agency with a total area of 400 km^2. The high-resolution WV-3 images contain eight (1.24 meters) spectral bands and a panchromatic band (0.3 meters). We applied the Gram-Schmidt pan-sharping method to obtain a high-resolution color image, conjugating the spatial information from the panchromatic band and spectral information from the multispectral bands.

Geographic Information System (GIS) specialists performed manual annotations considering fourteen distinct features, all listed in the table 8.1. Six of these classes were "things", and eight were "stuff" categories. The most numerous class was the straw umbrella, with nearly 4,000 distinct polygons and nearly no pixels with no classes. Figure 8.2 shows the annotation pattern, containing three examples of each interest class demarcated by a colored polygon.

The images were cropped into smaller image tiles with their corresponding annotations in the COCO format, which is the standard format for Detectron2's Panoptic-FPN model. These annotations require JSON files with specifications for each image, containing the information regarding the "thing" classes (such as bounding boxes) and "stuff" classes. The conversion of the GIS data to the panoptic format used the software developed by de Carvalho et al. (2022c), considering a GIS attribute table where each polygon has two columns with the class value and the polygon value (or unique IDs), in case of the thing category. The software requires point shapefiles to generate the smaller samples, in which each point is the centroid of the frame. We chose a sample size of 128x128 pixels. We used the image from 2017 to generate all training samples totaling 3,200. Using the image from 2018, we chose the validation and test samples, 300 of each. Choosing points manually is frequently better than random since we can select high-priority areas. Since the validation and test samples are in the same image, we assured that there was no overlap between them.

Table (8.1) Categories (in which SP, SBU, and TU stand for swimming pool, straw beach umbrella, and tourist umbrella), labels, type (thing or stuff), number of polygons, and number of pixels in the Panoptic Beach Dataset

Category	Label	type	polygons	pixels
Background	0	-	-	-
Ocean	1	stuff	-	47,799,957
W. Sand	2	stuff	-	3,874,445
Sand	3	stuff	-	7,944,837
Road	4	stuff	-	2,113,790
Vegetation	5	stuff	-	1,814,849
Grass	6	stuff	-	1,814,967
Sidewalk	7	stuff	-	1,360,113
Vehicle	8	thing	531	52,196
SP	9	thing	55	33,118
Construction	10	thing	457	1,097,025
SBU	11	thing	3,653	247,723
TU	12	thing	805	45,129
Crosswalk	13	thing	46	34,138

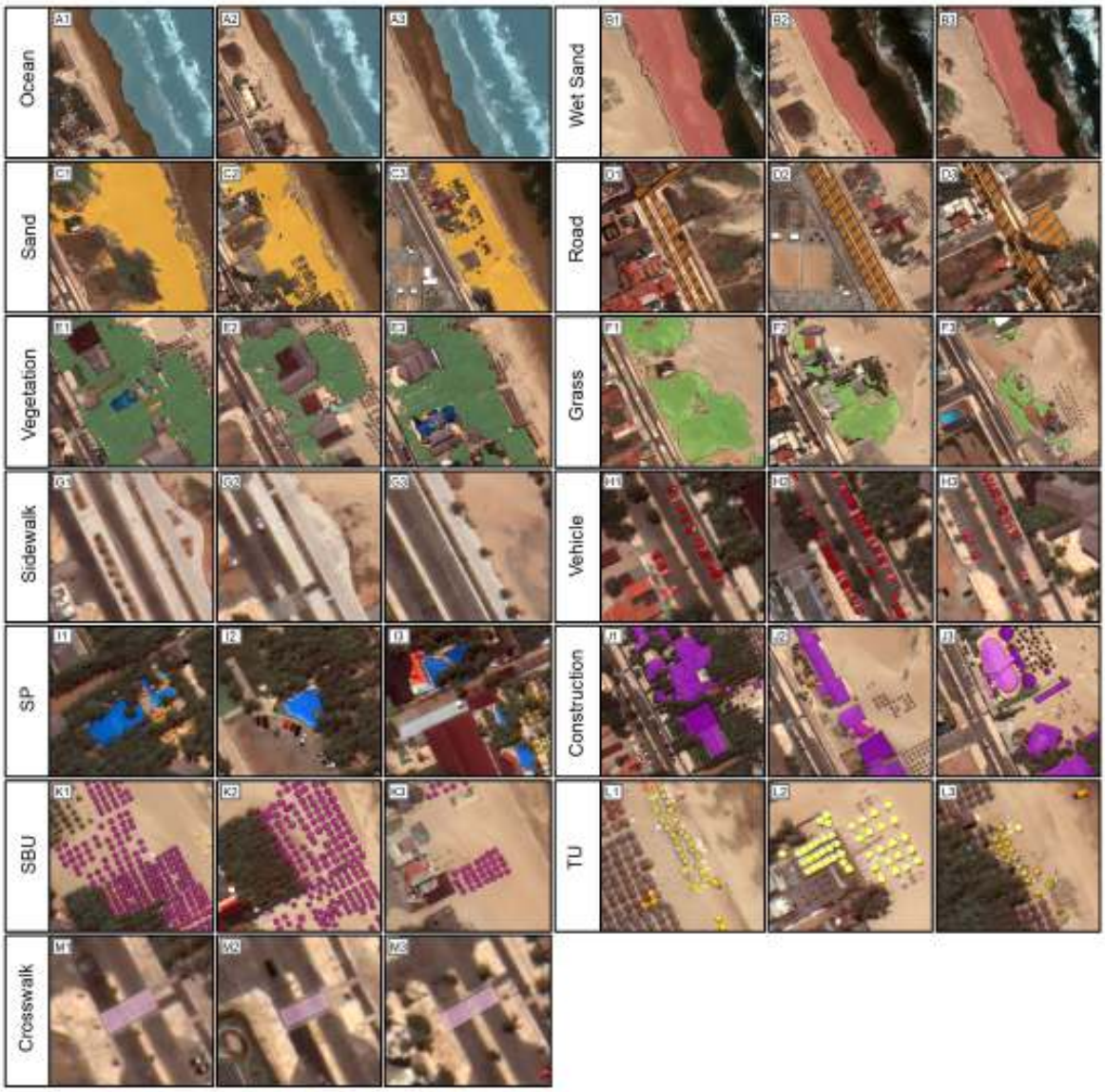

Figure (8.2) Examples of annotations for each class. The highlighted segments show the class corresponding to the written labels, in which we considered: Ocean, Wet Sand, Sand, Road, Vegetation, Grass, Sidewalk, Vehicle, Swimming Pool (SP), Construction, Straw Beach Umbrella (SBU), Tourist Umbrella (TU), and Crosswalk.

8.2.3 Deep Learning experiments

The experiments were subdivided into panoptic and semantic segmentation. The panoptic segmentation approach considered an analysis of different spectral band compositions using three models. Then, we evaluated our dataset using the semantic segmentation task considering 15 models for the best spectral band composition. Note that there is no need to perform an isolated instance segmentation approach since we can retrieve instance-only results from the panoptic models. All experiments were conducted on a computer with an NVIDIA RTX 3090 graphics card with 24GB RAM and an i9 processor.

Panoptic Segmentation

This research aims to compare different configurations of spectral bands using the Panoptic-FPN (Kirillov et al., 2019) model, the pioneer model in panoptic segmentation studies. The primary motivation for using FPN to predict semantic segmentation is to establish a simple, single-network baseline, which allows executing the semantic and instance segmentation steps in a chained way and considering a joint task of panoptic segmentation. This model is present in the Detectron2 software that allows implementation and contains detailed documentation for future improvements and replication. The software uses the Pytorch library, which is also widely used, making the code easier to understand. The Panoptic-FPN model comprises two branches: (1) instance segmentation and (2) semantic segmentation. Both branches use a common structure, which is the FPN. The instance segmentation branch uses a Mask-RCNN model and aims to identify the "things" elements (He et al., 2020). The semantic segmentation branch uses upsampling in the feature maps and targets the "stuff" classes. The two branches are combined using a simple heuristic method for combining the instance level "thing" predictions and the background "stuff" elements. The Detectron2 (Wu et al., 2019) software is the most appropriate to do experiments in this task because the documentation is very robust. Previous studies proposed modifications and adaptations for well functioning in remote sensing datasets (Carvalho et al., 2021; de Carvalho et al., 2022c), which has not yet been done to other models. We evaluated three backbones using the Panoptic-FPN model, namely the ResNeXt-101, ResNet-101, and ResNet-50.

We had to leverage the Detectron2 software for working with TIFF multispectral images, allowing it to use a varying number of input bands. Our experiments considered five tests from the pan-sharpening images, considering: (1) all eight spectral bands, (2) $RGB + NIR1 + NIR2$, (3) $RGB + NIR1$, (4) $RGB + NIR2$, and (5) only RGB. All trained models, apart from the input spectral dimensions, use the same specifications. The z-score normalization for each channel allowed a faster convergence in the training phase.

Regarding the model hyperparameters, we used: (a) stochastic gradient descent (SGD) optimizer; (b) 0.0005 learning rate; (c) 150,000 iterations; (d) anchor boxes with sizes 8, 16, 32, 64, 128; (e) three aspect ratios (0.5, 1, 2); (f) one image per batch. We evaluated the validation set for every 5,000 iterations, in which the final model considered the best Panoptic Quality results. Besides, we considered the following augmentation steps: (a) random vertical flip (probability chance of 50%), (b) random horizontal flip (probability chance of 50%), and (c) rescaling the image dimensions to 800x800 pixels. The augmentation processes in the training set resulted in 9,600 different image combinations in the training phase.

Semantic Segmentation

Even though the panoptic and semantic tasks present different objectives, the comparison is valid for analyzing the pros and cons of each approach. The semantic segmentation field has been much more explored in the remote sensing community, and the various models and implementations are better documented. The way we have constructed our dataset enables researchers to use

different tasks. In the semantic segmentation analysis, we compared five architectures (U-Net (Ronneberger et al., 2015b), U-Net++ (Zhou et al., 2018b), DeepLabv3+ (Chen et al., 2018), FPN (Lin et al., 2017), LinkNet (Chaurasia and Culurciello, 2017)) and three backbones (Efficient-net-B7 (Tan and Le, 2019b), ResNet-101 (He et al., 2016), and ResNeXt-101 (Xie et al., 2017)). All models considered the same loss function (cross-entropy) and hyperparameter settings, including 0.0005 learning rate, batch size of 25, Adam optimizer, and 300 epochs.

8.2.4 Accuracy Analysis

This study considers panoptic and semantic segmentation models that show remarkable differences. The Panoptic segmentation task involves three distinct types of evaluations: "stuff", "thing", and panoptic metrics. The per-pixel metrics suitable for semantic segmentation are the same as for the "stuff" evaluation. The difference is that the panoptic model only considers the "stuff" classes (for the stuff evaluation), and semantic segmentation considers all classes.

The "stuff" evaluation considered: (a) mean Intersection over Union, (b) frequency weighted IoU (fwIoU), (c) mean Accuracy (mAcc), and (d) pixel accuracy (pAcc). The mIoU corresponds to the mean average from all classes considering their area of intersection ($A \cap B$) divided by the area of union ($A \cup B$), in which A is the deep learning prediction and B is the ground truth. The fwIoU is similar but assigns weights according to the number of representations instead of a mean average. The pixel accuracy is simply the number of correctly classified pixels divided by the total number of pixels, and the mean Accuracy is the average among the accuracies from all different classes. Due to the differences between the types of segmentation, we define mIoU to denote the mean across all categories (semantic segmentation) and mIoU_{stuff} to denote the mean across the stuff categories (panoptic segmentation). The semantic segmentation evaluation considered the mIoU and mIoU_{stuff} metrics.

In the thing evaluation (instance segmentation), the COCO Average Precision (AP) is the primary metric not only in the COCO challenge (Lin et al., 2014) but also in many studies (Bolya et al., 2020; Cai and Vasconcelos, 2018; Gao et al., 2021; He et al., 2020; Huang et al., 2019). The AP is a ranking metric expressed as the area under the precision-recall curve. In order to calculate precision and recall, we must identify the correctly predicted elements. The COCO metric uses different IoU thresholds between the predicted and ground truth bounding boxes. The primary AP metric considers 10 IoU thresholds, from 0.5 to 0.95, with 0.05 steps. To exclusively evaluate a more and less strict version, the AP50 and AP75 use the 0.5 and 0.75 thresholds.

Finally, the Panoptic metrics are: Panoptic Quality (PQ), Segmentation Quality (SQ), and Recognition Quality(de Carvalho et al., 2022c; Gao et al., 2021; Kirillov et al., 2019; Mohan and Valada, 2021a). The PQ is the primary metric for this task, being the multiplication of the SQ by RQ, where SQ is $\frac{\sum_{(pred,GT)\in TP} IoU(pred,GT)}{|TP|}$ and RQ is $\frac{TP}{|TP|+\frac{1}{2}|FP|+\frac{1}{2}|FN|}$, in which pred, GT, TP, FN, and FP stands for the deep learning prediction, ground truth data, true positives (elements with an IoU greater than 0.5), false negatives, and false positives, respectively.

8.3 Results

This section is subdivided in three parts: (1) panoptic segmentation evaluation, (2) semantic segmentation evaluation, and (3) visual results that show the differences of semantic, instance, and panoptic segmentation.

8.3.1 Panoptic Segmentation evaluation

Panoptic metrics

Table 8.2 lists the panoptic segmentation results for the PQ, SQ, and RQ metrics. Our models were selected from the validation set based on the best PQ performance, being the main metric for evaluation. ResNeXt-101 was the best backbone for all band compositions, followed by ResNet-101. Using all spectral bands provided the best PQ, SQ, and RQ results. The composition with only RGB bands obtained a significantly lower accuracy, showing that the NIR1 and NIR2 bands are significant for classifying typical targets in the beach scenario.

Table (8.2) Panoptic Quality (PQ), Segmentation Quality (SQ), and Recognition Quality (RQ) results for the ResNet-50, ResNet-101, and ResNeXt-101 backbones. The best results are in bold.

Spectral bands	PQ	SQ	RQ
ResNeXt-101			
All	**65.90**	**81.23**	**80.83**
RGB+NIR1+NIR2	65.43	80.90	80.49
RGB+NIR1	64.82	80.67	79.94
RGB+NIR2	64.37	80.65	79.50
RGB	61.23	79.32	76.89
ResNet-101			
All	64.88	79.51	80.80
RGB+NIR1+NIR2	64.60	79.89	80.40
RGB+NIR1	64.41	80.10	79.94
RGB+NIR2	64.30	80.87	79.13
RGB	61.21	79.45	76.66
ResNet-50			
All	63.04	78.54	79.71
RGB+NIR1+NIR2	62.65	79.52	78.33
RGB+NIR1	62.52	78.91	78.73
RGB+NIR2	62.20	79.90	77.48
RGB	60.87	79.18	76.51

Stuff evaluation results

Table 8.3 lists the macro results for the stuff categories. In contrast to the panoptic metrics, using all bands did not yield the maximum results, even though they were very close. The panoptic models use a loss function that considers many elements, and there may be a tradeoff between some of them to yield the best results The RGB-only composition was considerably lower than the rest. When analyzing this behavior per class (Table 8.4), Wet Sand, Road, and Grass classes presented considerably lower results. The difference of RGB-only metrics considering the other classes did not show much difference, showing that depending on the classes being evaluated,

the RGB bands can be satisfactory. Moreover, an exciting result is that most of the classes showed values above 80% in IoU, showing that the targets in the beach setting are suitable for expanding monitoring with deep learning methods.

Table (8.3) Metric analysis for the "stuff" categories, considering Mean Intersection over Union ($mIoU_{stuff}$), frequency weighted ($fwIoU_{stuff}$), mean accuracy ($mAcc_{stuff}$), and pixel accuracy ($pAcc_{stuff}$) results for semantic segmentation in the Beach dataset. The best results for each class are in bold.

spectral bands	$mIoU_{stuff}$	$fwIoU_{stuff}$	$mAcc_{stuff}$	$pAcc_{stuff}$
ResNeXt-101-32x8d				
8 bands	85.35	88.48	91.88	93.74
RGB+NIR1+NIR2	**85.58**	88.63	**92.02**	93.84
RGB+NIR1	85.10	88.34	91.91	93.63
RGB+NIR2	84.32	87.65	91.24	93.29
RGB	81.49	85.89	89.88	92.17
ResNet-101				
8 bands	84.76	88.23	91.41	93.57
RGB+NIR1+NIR2	85.53	88.78	91.93	93.90
RGB+NIR1	85.57	**88.83**	91.96	**93.93**
RGB+NIR2	85.10	88.47	91.92	93.73
RGB	82.27	86.44	90.02	92.54
ResNet-50				
8 bands	84.02	87.37	90.93	93.08
RGB+NIR1+NIR2	84.31	87.57	91.32	93.21
RGB+NIR1	83.58	86.88	91.47	92.74
RGB+NIR2	82.88	86.53	90.62	92.56
RGB	82.24	86.10	90.19	92.33

Table (8.4) Intersection over Union (IoU) results for the "stuff" categories per class in the Beach dataset, in which (1) ocean, (2) Wet Sand, (3) Sand, (4) Road, (5) Vegetation, (6) Grass, and (7) sidewalk. The best results for each class are in bold.

spectral bands	1	2	3	4	5	6	7
ResNeXt-101							
All	96.40	87.72	92.49	**90.09**	83.84	74.00	76.54
RGB+NIR1+NIR2	96.73	89.04	92.46	89.48	83.74	**74.48**	**77.06**
RGB+NIR1	97.69	90.52	92.14	88.97	82.50	72.32	76.00
RGB+NIR2	95.06	85.20	92.24	89.19	83.56	72.46	76.05
RGB	94.93	80.84	91.49	83.29	82.48	63.62	74.75
ResNet-101							
All	**98.64**	90.91	91.83	87.01	82.34	71.31	74.13
RGB+NIR1+NIR2	98.63	**92.16**	92.22	87.01	83.39	72.90	75.83
RGB+NIR1	97.95	91.31	**92.71**	87.57	83.40	73.23	76.79
RGB+NIR2	96.78	88.06	92.65	87.46	**84.62**	73.39	75.92
RGB	96.33	84.21	91.67	80.97	82.40	69.60	72.54
ResNet-50							
All	97.96	90.30	90.97	85.81	81.84	72.19	73.81
RGB+NIR1+NIR2	97.79	87.87	91.05	86.04	82.82	73.70	75.21
RGB+NIR1	97.04	87.94	90.53	86.21	82.57	73.10	73.21
RGB+NIR2	95.74	86.29	91.34	85.44	82.56	71.10	73.51
RGB	95.51	84.67	90.94	83.06	82.59	68.57	74.54

Thing evaluation results

Table 8.5 lists the COCO metrics (AP, AP_{50}, AP_{75}) results for the "thing" classes. The ResNeXt-101 presented the best values for all combinations considering the AP metric. Even though the ResNet-101 and ResNet-50 presented the worst values for the main metric, the AP_{50} was superior for many configurations. The influence of band composition on the things classes was much less significant, in which none of the classes had a significantly worst behavior. The main factor for results for the thing classes is the backbone.

Table (8.5) COCO metrics for the thing categories in the Beach Dataset considering the usage of different spectral bands: (1) all (eight spectral bands), (2) Red, Green, and Blue (RGB) with NIR1 and NIR2, (3) RGB with NIR1, (4) RGB with NIR2, and (5) only RGB. The best results for Box and Mask are in bold.

Spectral bands	Type	AP	AP_{50}	AP_{75}
ResNeXt-101				
All	Box	**60.05**	83.66	59.51
	Mask	53.39	82.23	55.35
RGB+NIR1+NIR2	Box	59.31	83.00	**63.21**
	Mask	**54.67**	81.85	56.38
RGB+NIR1	Box	59.48	82.89	66.35
	Mask	54.52	79.92	**59.30**
RGB+NIR2	Box	59.11	83.43	60.61
	Mask	54.19	81.82	55.45
RGB	Box	59.52	83.27	64.74
	Mask	53.70	81.19	58.70
ResNet-101				
All	Box	57.30	87.22	61.35
	Mask	50.49	85.18	54.05
RGB+NIR1+NIR2	Box	56.69	**87.30**	60.65
	Mask	51.23	**85.52**	54.23
RGB+NIR1	Box	58.38	85.01	61.96
	Mask	52.36	83.56	54.84
RGB+NIR2	Box	57.52	82.93	61.34
	Mask	53.13	80.25	58.95
RGB	Box	56.34	83.84	61.77
	Mask	48.97	82.55	50.82
ResNet-50				
All	Box	51.54	86.31	52.78
	Mask	46.24	83.92	45.30
RGB+NIR1+NIR2	Box	52.72	85.35	55.78
	Mask	44.87	83.13	45.77
RGB+NIR1	Box	50.85	82.67	53.78
	Mask	46.09	80.49	50.61
RGB+NIR2	Box	51.10	83.51	53.15
	Mask	45.82	82.41	49.62
RGB	Box	50.21	82.53	53.13
	Mask	44.43	78.60	47.84

Table (8.6) COCO metrics for the thing categories in the Beach Dataset considering the usage of different spectral bands: (1) all (eight spectral bands), (2) Red, Green, and Blue (RGB) with NIR1 and NIR2, (3) RGB with NIR1, (4) RGB with NIR2, and (5) only RGB. The evaluated classes are: (8) vehicle, (9) SP, (10) construction, (11) straw beach umbrella, (12) tourist umbrella, and (13) crosswalk. The best results for Box and Mask are in bold.

Spectral bands	Type	8	9	10	11	12	13
ResNeXt-101							
All	Box	60.67	58.23	56.75	39.42	74.25	70.72
	Mask	54.47	49.17	54.61	33.75	63.65	64.71
RGB+NIR1+NIR2	Box	61.36	59.61	58.38	39.18	73.65	63.68
	Mask	56.31	49.71	57.13	35.38	66.80	62.68
RGB+NIR1	Box	61.05	57.43	58.38	35.58	71.19	73.26
	Mask	55.39	47.89	56.98	33.20	63.62	70.05
RGB+NIR2	Box	58.85	57.56	57.93	37.76	72.55	70.01
	Mask	53.98	46.84	55.56	34.38	67.70	66.65
RGB	Box	61.96	56.71	58.09	39.62	70.06	70.73
	Mask	55.10	46.03	55.50	35.86	65.96	63.77
ResNet-101							
All	Box	59.70	54.96	58.15	47.16	58.17	65.63
	Mask	52.63	44.63	54.69	41.21	53.52	56.22
RGB+NIR1+NIR2	Box	60.14	53.28	59.02	**48.74**	61.69	57.27
	Mask	52.38	45.91	57.13	**45.03**	55.84	51.12
RGB+NIR1	Box	58.44	56.08	58.84	44.65	66.00	**66.26**
	Mask	52.06	49.55	**58.00**	40.32	59.06	55.23
RGB+NIR2	Box	59.06	**56.42**	**59.68**	45.18	69.99	54.79
	Mask	53.02	50.66	56.90	41.44	63.45	53.34
RGB	Box	59.23	58.94	55.48	45.23	55.12	64.03
	Mask	49.36	50.45	53.86	42.08	47.51	50.55
ResNet-50							
All	Box	59.68	51.26	53.09	40.52	46.49	58.17
	Mask	51.74	42.16	50.50	32.28	40.12	**59.63**
RGB+NIR1+NIR2	Box	57.08	53.47	53.81	37.43	51.90	62.65
	Mask	48.82	41.97	50.68	31.94	43.65	52.15
RGB+NIR1	Box	59.08	55.46	52.16	29.12	49.89	59.37
	Mask	**53.84**	48.90	50.27	26.97	43.47	53.09
RGB+NIR2	Box	**62.48**	49.46	54.28	35.48	49.13	55.75
	Mask	53.51	48.18	52.01	28.31	41.63	51.28
RGB	Box	59.99	49.04	52.97	32.85	47.77	58.65
	Mask	50.47	44.34	49.38	28.98	37.70	55.60

8.3.2 Semantic segmentation results

This section shows the benchmark results for the Beach Dataset considering the semantic segmentation task. Table 8.7 lists the mIoU metrics for each model. Note that the metrics shown here are different from the mIoU_{stuff} since we are now considering all classes. For an easier comparison of this section with the panoptic models, we also incorporated the mIoU_{stuff}. The FPN model presented the highest mIoU (77.44) and mIoU_{stuff} (85.67) results. These results are slightly higher than the mIoU_{stuff} from the best panoptic model (85.58). Within the different architectures, the Efficient-net-B7 was the best backbone, followed by ResNeXt-101. The differences across architectures was smaller than across backbones.

Table (8.7) Semantic segmentation model metrics considering all spectral brands. The best results are in bold.

Architecture	Backbone	mIoU	$mIoU_{stuff}$
U-Net	Eff-B7	75.56	85.63
	ResNeXt-101	73.25	84.05
	ResNet-101	71.49	83.19
DeepLabv3+	Eff-B7	75.68	85.44
	ResNeXt-101	73.67	83.95
	ResNet-101	71.73	83.17
U-Net++	Eff-B7	75.71	85.35
	ResNeXt-101	73.47	83.84
	ResNet-101	69.13	82.09
LinkNet	Eff-B7	74.74	84.42
	ResNeXt-101	73.78	83.45
	ResNet-101	69.11	81.28
FPN	Eff-B7	**77.44**	**85.67**
	ResNeXt-101	73.52	84.21
	ResNet-101	72.61	83.17

8.3.3 Visual Results

Figure 8.3 shows five examples from the test set considering the original image, and the panoptic, instance, and semantic predictions. The concept of panoptic segmentation changes the presentation configuration where each stuff category has a unique color, while thing categories has unique values, and consequently colors for each object. Therefore, this technique brings a new approach to the cartographic representation of land use/land cover maps, which usually adopts a pixel classification. The Panoptic-FPN model generates a JSON for each predicted image, retrieving more attributes of each element, such as the bounding box and the class, favoring other ways of visualizing the data. We removed the bounding boxes for visual purposes, as overlapping information would make the image cluttered. Even though some metrics do not seem very high, mainly related to the nature of small objects, the model can predict crowded correctly and numerous elements in a single image from a visual perspective. The results show that "thing" targets close to each other tend to merge the predictions, making it very difficult to separate different instances, as shown in Fig 8.3A3 and E3. The beach setting has many amorphous elements, which are disconsidered by the instance predictions, demonstrating that the panoptic segmentation aggregates the benefits of both methods.

8.4 Discussion

This research adapted the original Panoptic-FPN code of the panoptic segmentation to perform the joint processing of all available multispectral bands and to couple the ResNeXt-101 backbone, considering multiclass "thing" and "stuff". The panoptic segmentation task presents a much larger complex in the model design compared to the instance and semantic segmentation task. The loss function encompasses both instance and semantic segmentation losses. The instance segmentation also presents a complexity since it involves segmentation loss, bounding box regression loss, and classification loss. All of those elements associated with a large number

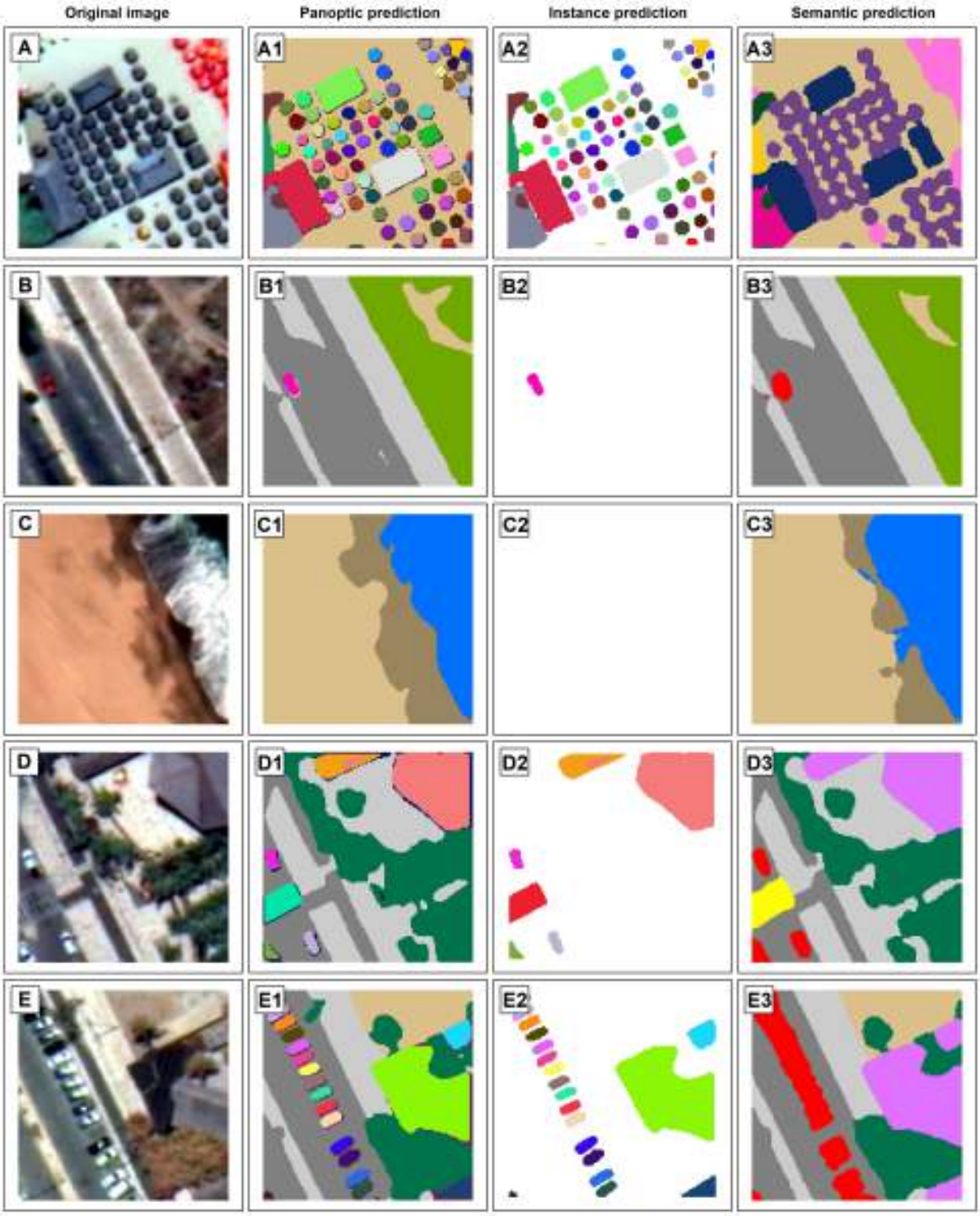

Figure (8.3) Six examples with the original image (considering the RGB bands) and the prediction. The predictions maintained the same colors for the stuff classes, and each thing class presents a varying color.

of classes may present fluctuations when comparing the models. For example, the AP metrics for the individual targets may show a higher metric for a single target given a model, this is why it is essential to analyze the macro metrics since they provide an overall view of the model,

and they should primarily focus on identifying and building the best model.

The investigation used the Panoptic-FPN model of Detectron2 software proposed by Meta Artificial Intelligence Research and used and tested globally. Unlike semantic segmentation, which has several models developed, panoptic segmentation has a restricted number of models that still lack detailed documentation. We compared the Panoptic-FPN architecture with three different backbones (ResNeXt-101, ResNet-101, and ResNet-50). This study evaluated different compositions of spectral bands within the panoptic segmentation. This approach allows quantifying the gain in accuracy with the use of multispectral data. The Panoptic-FPN model using ResNeXt-101 backbone and all bands obtained better results in all panoptic metrics (PQ, SQ, and RQ). Even though the panoptic segmentation results were similar using all bands, $RGB + NIR1 + NIR2$, $RGB + NIR1$, and $RGB + NIR2$, the spectral characteristics from remotely sensed data can enhance the results significantly when compared to the traditional RGB channels with nearly 5% worse than the rest in the PQ. This deep learning survey was also the first to use multiple remote sensing targets in the beach setting.

The results can guide other researchers in selecting bands for future studies in a beach scenario. Similar studies achieved some complementary results to our findings. Carvalho et al. (2021) compared RGB and all bands from the Landsat-8 sensor for center pivot mapping using instance segmentation models, where the authors found that using all bands had a 3% increase in the metric AP. In a semantic segmentation study, Barros et al. (2022) found that the NIR band was almost sufficient to map vineyards. Furthermore, the remote sensing field has many opportunities for studies using multichannel inputs, especially considering time series and multispectral data (Carvalho et al., 2021; de Albuquerque et al., 2021b). Recent studies have used a time-series sequence as the input, in which each time represents a different channel (de Albuquerque et al., 2021a; de Bem et al., 2021; Li et al., 2020b). In many of these studies, we can see that introducing new information is complementary to deep learning studies until it reaches the point when new information is redundant. This significant analysis allows primary bands to be selected instead of all available bands, reducing the computational cost.

However, the panoptic segmentation task is challenging to compare with other deep learning and machine learning methods because the evaluation criteria are very different from other methodologies, such as instance and semantic segmentation that do not have the categories together of "stuff" and "thing." Recently, De Carvalho et al. (2022) proposed a novel way to approach panoptic segmentation with semantic segmentation models, which could also be an alternative way to address a more robust model comparison in future studies. Despite the difficulty in a metric-wise comparison, we can evaluate the benefits of each task, especially when using a visual comparison. As shown in the results section, the IoU results for semantic segmentation models are very accurate for some classes. However, the beach targets are crowded in many cases, such as the beach straw umbrellas (the most numerous category). For targets like this, it is very hard to retrieve relevant information, such as the number of individual elements, since the semantic predictions tend to aggregate many of the targets that are close to each other, similar to what happens in vehicle detection (de Carvalho et al., 2022a; Mou and Zhu, 2018). On the other hand, the instance segmentation also has limitations to important classes such

as sand, sidewalks, and roads. The panoptic segmentation emerges as a viable and interesting solution for handling the beach setting with many objects and backgrounds. Our novel proposed dataset presents different characteristics than other panoptic segmentation datasets, formed by RGB and ground-level images (Cityscapes, COCO, and Mapillary Vistas). Although the BSB Aerial Dataset (de Carvalho et al., 2022c) consists of aerial photos for panoptic segmentation, the available channels are RGB. The present dataset contains orbital multispectral images, considering images composed of up to 8 bands from the WV-3 sensor. Changing the RGB inputs to include all available spectral bands produces better results. Using ground-level RGB dataset transfer learning for multispectral imaging can still provide some leverage to reduce the training period as low-level features such as corners are similarly represented. Even so, transfer learning between different sensor data sets is complicated as each sensor has different amounts of spectral bands with different characteristics of the spectrum range, providing less accurate transferability. A possible solution and future studies would include building a dataset using various sensors simultaneously, covering a wide range of spectral and spatial behaviors.

Most of the classes are in the range of small objects. The small objects are a great difficulty since their representation is much less significant, bringing difficulties for classification. In many datasets such as COCO, the AP_{small} metrics is much lower than the rest. The results may be affected by the size of the objects. In a previous study in this same region, de Carvalho et al. (2021d) analyzed only the class Straw Beach Umbrella with different scaling dimensions, in which they upscaled the image up to 8 times the original size. The AP metric nearly doubled by a simple operation. One of the augmentation steps in our research was to resize the image to 800x800 spatial dimensions, a typical and default setting in the Detectron2 software. The nature of the metrics is not favorable for achieving high results since small displacements in the bounding boxes significantly affect the metric, which does not happen for larger objects. Tong et al. (2020) made a review article on small objects, in which they stated that increasing the dimensions is one of the simplest forms of increasing results. Kisantal et al. (2019) created a method for increasing the representation of small objects. Even though this solution is up-and-coming and can indeed increase the results, in some situations, the fact that the objects are small is enough to bring the metric down, and sometimes in visual results, the prediction is very accurate.

Finally, the results proved to be entirely satisfactory in the landscape of Praia do Futuro, an important area for government inspection for having high tourist activity on public lands. Therefore, the model has immediate application in the periodic monitoring of this urban beach with constant misuse of public property. However, a limitation of the present investigation is that the model is only suitable for beaches with similar characteristics (same composition of sediments, vegetation, and tourist infrastructure). Therefore, future research should include various beach settings, regions, and countries. In addition, future research may test other images, such as drones.

8.5 Conclusion

This study pioneered the panoptic segmentation tasks in the beach setting, considering high-resolution WorldView-3 images with 0.31-meter resolution and a multispectral dataset with "things" and "stuff" classes. Since most computer vision developments use RGB image datasets, we evaluated different configurations of band arrangements and found that the combination of the near infra-red (NIR) and the RGB bands can significantly improve results. The beach setting's main panoptic metric (PQ) differed by nearly 5%. The difference in using all eight multispectral bands with $RGB + NIR1$ and NIR2 is very shallow, and in situations with fewer computational resources, using fewer bands will not affect the results. The Panoptic-FPN architecture with the ResNeXt-101 backbone performed better for panoptic metrics than the ResNet-101 and ResNet-50.

The panoptic segmentation presents a new possibility for developing inspection solutions in the beach areas. We can simultaneously retrieve crucial information about individual objects, such as their size and abundance. Future research aims to develop methodologies for mapping large regions, encompassing sliding window methods, which are still unexplored for panoptic segmentation, and using other images such as aerial images and unmanned aerial vehicles.

Chapter 9

Concluding Remarks

The present dissertation assessed different segmentation methods based on deep learning (semantics, instance, and panoptic) applied to remote sensing data (RGB and multispectral images), embracing a wide variety of models. Furthermore, the dissertation proposed new methodological approaches considering instance and panoptic segmentation methods with and without bounding boxes.

The first analysis of the dissertation evaluated box-based instance segmentation methods (Mask-RCNN) for detecting small objects observed on multispectral images. One of the most significant difficulties among box-based methods is predicting extensive areas since the deep learning samples are often much smaller. In this regard, only recently, a proposition was made using sliding windows using non-maximum suppression. This dissertation proposes a method by incorporating the double edge classifier (DEG), reducing the number of possible errors in the final image. Still using box-based methods, the topic regarding small objects is very relevant yet little explored. Therefore, image scaling can be a straightforward solution rather than optimizing many parameters and other settings. The AP metric improved nearly 100% by increasing the image dimensions eight times, but doubling the image already has significant improvements. This study was very relevant since all practical applications using box-based methods would require some methodology for large image classification, which can guide practical results and other researchers for enhancing this method.

Even though we have built a successful box-based method for small objects, semantic segmentation models are better at a pixel level and more straightforward for training and classifying large regions. This limitation happens because the mask in each region of interest has reduced dimensions (28x28) which compromises an exact delineation. On the other hand, semantic segmentation models are very precise at a pixel level but present difficulties in recognizing unique objects since cluttered predictions tend to merge (representation of multiple elements by a single polygon). To make the most out of semantic segmentation models' accuracy, we proposed a novel box-free method that considers the object's borders. Borders can isolate the objects' interiors, assigning different values to each prediction and enabling us to achieve instance segmentation results with non-learning methods. The evaluation of this procedure considered the vehicle class, which has many examples, and its manual annotation is highly labor-intensive. In

this regard, the present research proposes the first pipeline using a semi-supervised approach with GIS and deep learning. The proposed method creates datasets quickly and efficiently from the separation of a set of tests that evaluates the convergence of the model, suitable for any other remote sensing target. Our dataset contained more than 120 thousand different vehicles, which are considered the most crucial zones of Brasilia.

Panoptic segmentation is the latest method that provides a complete understanding of the scene, suppressing the main difficulties of semantic and instance segmentation and obtaining instance-level predictions for objects while still mapping the background classes. However, no study used panoptic segmentation in the remote sensing field. One of the main reasons for this lack of research is the difficulty in building panoptic segmentation datasets due to the quantity of information required. The panoptic data requires box information and semantic information. The most conventional is the COCO annotation format which is still very little friendly for other users. Thus, we have built software that gathers GIS data and transforms it into the COCO panoptic format. Our methodology requires a few easy steps in any GIS database by providing some information in the attribute tables. This research created the first panoptic dataset using remote sensing data with software development and pipeline considering fourteen classes. GIS data and vectors have been used very long before the rise of deep learning, and our proposed method enables other researchers to quickly transform databases to the panoptic format and use panoptic segmentation models such as the Panoptic-FPN. This first study considered aerial RGB images, and to extend this work to different remote sensing data, we applied the same methodology in the beach setting using multispectral data. Further research on this topic includes augmenting this dataset with more samples and making a more thorough comparison to see the importance of the multispectral bands in the Panoptic-FPN model.

Apart from the difficulty in the data generation, the remote sensing community is more adapted for using semantic segmentation models than instance and panoptic. The semantic segmentation models have a more straightforward data preparation process since they only require the ground truth image. This facility makes the use of semantic segmentation ten times greater than instance segmentation. We also proposed a box-free solution for obtaining panoptic predictions. In top-view images, some elements can touch others from the same class (e.g., cars), but other classes never touch each other (e.g., swimming pools). Therefore, there are two categories to isolate the objects considering touching objects and non-touching objects. Our solution extended the vehicles' methodology and proposed the first sliding window approach for panoptic predictions. This research can be handy in practical scenarios since the sliding windows approach is simplified.

Finally, most datasets consider a modal perspective, i.e., the annotations are done only in the visible parts of the images. The amodal approach to identifying non-visible parts of objects has not yet been applied in remote sensing. In future studies, we aim to adopt the semantic, instance, and panoptic segmentation to the amodal perspective.

References

Abdollahi, A., Pradhan, B., Gite, S., and Alamri, A. (2020). Building footprint extraction from high resolution aerial images using generative adversarial network (gan) architecture. *IEEE Access*, 8:209517–209527. 73

Albawi, S., Mohammed, T. A., and Al-Zawi, S. (2017). Understanding of a convolutional neural network. In *2017 international conference on engineering and technology (ICET)*, pages 1–6. Ieee. 8

Albuquerque, A. O., de Carvalho, O. L. F., e Silva, C. R., de Bem, P. P., Trancoso Gomes, R. A., Borges, D. L., Guimarães, R. F., Pimentel, C. M. M., and de Carvalho Júnior, O. A. (2021a). Instance segmentation of center pivot irrigation systems using multi-temporal sentinel-1 sar images. *Remote Sensing Applications: Society and Environment*, 23(March):100537. 14, 24

Albuquerque, A. O., Ferreira de Carvalho, O. L., Silva, C., Saiaka Luiz, A., De Bem, P. P., Gomes, R. A. T., Guimaraes, R. F., and Decarvalhojunior, O. A. A. (2021b). Dealing with clouds and seasonal changes for center pivot irrigation systems detection using instance segmentation in sentinel-2 time series. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, pages 1–1. 24

Aly, M. (2005). Survey on multiclass classification methods. *Neural Netw*, 19(1-9):2. 9

Ammar, A., Koubaa, A., Ahmed, M., Saad, A., and Benjdira, B. (2021). Vehicle detection from aerial images using deep learning: A comparative study. *Electronics (Switzerland)*, 10(7):1–31. 52

Ammour, N., Alhichri, H., Bazi, Y., Benjdira, B., Alajlan, N., and Zuair, M. (2017). Deep learning approach for car detection in uav imagery. *Remote Sensing*, 9(4):312. 49, 73

Audebert, N., Boulch, A., Randrianarivo, H., Le Saux, B., Ferecatu, M., Lefevre, S., and Marlet, R. (2017a). Deep learning for urban remote sensing. *2017 Joint Urban Remote Sensing Event, JURSE 2017*. 19, 25, 51, 69

Audebert, N., Le Saux, B., and Lefèvre, S. (2017b). Segment-before-detect: Vehicle detection and classification through semantic segmentation of aerial images. *Remote Sensing*, 9(4):368. 73

Azimi, S. M., Bahmanyar, R., Henry, C., and Kurz, F. (2021). Eagle: Large-scale vehicle detection dataset in real-world scenarios using aerial imagery. pages 6920–6927. IEEE. 69

Ball, J. E., Anderson, D. T., and Chan, C. S. (2017). Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community. *Journal of Applied Remote Sensing*, 11(04):1. 14

Barros, T., Conde, P., Gonçalves, G., Premebida, C., Monteiro, M., Ferreira, C. S., and Nunes, U. (2022). Multispectral vineyard segmentation: A deep learning comparison study. *Computers and Electronics in Agriculture*, 195:106782. 119

Bashir, S. M. A. and Wang, Y. (2021). Small object detection in remote sensing images with residual feature aggregation-based super-resolution and object detector network. *Remote Sensing*, 13(9):1854. 52

Belal, A. and Moghanm, F. (2011). Detecting urban growth using remote sensing and gis techniques in al gharbiya governorate, egypt. *The Egyptian Journal of Remote Sensing and Space Science*, 14(2):73–79. 12

Benedek, C., Descombes, X., and Zerubia, J. (2012). Building development monitoring in multitemporal remotely sensed image pairs with stochastic birth-death dynamics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(1):33–50. 91

Bengio, Y., Goodfellow, I., and Courville, A. (2017). *Deep learning*, volume 1. MIT press Cambridge, MA, USA. 4, 5

Benjdira, B., Khursheed, T., Koubaa, A., Ammar, A., and Ouni, K. (2019). Car detection using unmanned aerial vehicles: Comparison between faster r-cnn and yolov3. pages 1–6, Muscat, Oman. IEEE. 52

Bharathi, B. and Shamily, P. B. (2020). A review on iris recognition system for person identification. *International Journal of Computational Biology and Drug Design*, 13(3):316. 1, 14

Bini, S. A. (2018). Artificial intelligence, machine learning, deep learning, and cognitive computing: what do these terms mean and how will they impact health care? *The Journal of arthroplasty*, 33(8):2358–2361. 4

Bokhovkin, A. and Burnaev, E. (2019). Boundary loss for remote sensing imagery semantic segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11555 LNCS(14):388–401. 73

Bolya, D., Zhou, C., Xiao, F., and Lee, Y. J. (2019). Yolact: Real-time instance segmentation. Number May, pages 9156–9165, Seoul, Korea (South). IEEE. 73

Bolya, D., Zhou, C., Xiao, F., and Lee, Y. J. (2020). Yolact++: Better real-time instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1. 73, 84, 107, 112

Bonde, U., Alcantarilla, P. F., and Leutenegger, S. (2020). Towards bounding-box free panoptic segmentation. In *DAGM German Conference on Pattern Recognition*, pages 316–330. Springer. 11

Bradbury, K., Saboo, R., Johnson, T. L., Malof, J. M., Devarajan, A., Zhang, W., Collins, L. M., and Newell, R. G. (2016). Distributed solar photovoltaic array location and extent dataset for remote sensing object identification. *Scientific Data*, 3(May):1–9. 30

Bradski, G. and Kaehler, A. (2008). *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Inc. 56

Brasil, de Minas e Energia, M., and de Pesquisa Energética, E. (2022). *Plano Decenal de Expansão de Energia 2031*. MME/EPE, Brasilia. 29

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32. 8

Bro, A. S., Moran, E., and Calvi, M. F. (2018). Market participation in the age of big dams: The belo monte hydroelectric dam and its impact on rural agrarian households. *Sustainability (Switzerland)*, 10(5). 29

Brown, G., de Bie, K., and Weber, D. (2015). Identifying public land stakeholder perspectives for implementing place-based land management. *Landscape and Urban Planning*, 139:1–15. 12, 13

Brown, G., Weber, D., and de Bie, K. (2014). Assessing the value of public lands using public participation gis (ppgis) and social landscape metrics. *Applied Geography*, 53:77–89. 12

Burak, S., Dogan, E., and Gazioglu, C. (2004). Impact of urbanization and tourism on coastal environment. *Ocean Coastal Management*, 47(9-10):515–527. 13

Caesar, H., Uijlings, J., and Ferrari, V. (2018). Coco-stuff: Thing and stuff classes in context. pages 1209–1218, Salt Lake City, UT, USA. IEEE. 9

Cai, Z. and Vasconcelos, N. (2018). Cascade r-cnn: Delving into high quality object detection. pages 6154–6162, Salt Lake City, UT, USA. IEEE. 84, 112

Calvi, M. F., Moran, E. F., Silva, d. R. F. B., and Batistella, M. (2020). The construction of the belo monte dam in the brazilian amazon and its consequences on regional rural labor. *Land Use Policy*, 90(September 2019):104327. 29

Cao, L., Jiang, Q., Cheng, M., and Wang, C. (2016). Robust vehicle detection by combining deep features with exemplar classification. *Neurocomputing*, 215:225–231. 49

Cao, L., Luo, F., Chen, L., Sheng, Y., Wang, H., Wang, C., and Ji, R. (2017). Weakly supervised vehicle detection in satellite images via multi-instance discriminative learning. *Pattern Recognition*, 64(September 2016):417–424. 50

Cao, X., Wu, C., Lan, J., Yan, P., and Li, X. (2011). Vehicle detection and motion analysis in low-altitude airborne video under urban environment. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(10):1522–1533. 49, 50

Carvalho, d. O. L. F., de Carvalho Júnior, O. A., Albuquerque, d. A. O., Bem, d. P. P., Silva, C. R., Ferreira, P. H. G., Moura, d. R. d. S., Gomes, R. A. T., Guimarães, R. F., and Borges, D. L. (2021). Instance segmentation for large, multi-channel remote sensing imagery using mask-rcnn and a mosaicking approach. *Remote Sensing*, 13(1):39. 16, 19, 24, 25, 58, 71, 94, 97, 106, 111, 119

Carvalho, O., Albuquerque, A., Santana, N., de Carvalho Júnior, O., Gomes, R., Guimarães, R., and Borges, D. (2022). Bsb vehicle dataset. 64

Castro-Diaz, L., Lopez, M. C., and Moran, E. (2018). Gender-differentiated impacts of the belo monte hydroelectric dam on downstream fishers in the brazilian amazon. *Human Ecology*, 46(3):411–422. 28

Cha, J.-Y., Yoon, H.-I., Yeo, I.-S., Huh, K.-H., and Han, J.-S. (2021). Panoptic segmentation on panoramic radiographs: Deep learning-based segmentation of various structures including maxillary sinus and mandibular canal. *Journal of Clinical Medicine*, 10(12):2577. 106

Chaurasia, A. and Culurciello, E. (2017). Linknet: Exploiting encoder representations for efficient semantic segmentation. In *2017 IEEE Visual Communications and Image Processing (VCIP)*, pages 1–4. IEEE. 57, 112

Chen, C., Zhong, J., and Tan, Y. (2019). Multiple-oriented and small object detection with convolutional neural networks for aerial image. *Remote Sensing*, 11(18):2176. 52

Chen, L.-C., Papandreou, G., Schroff, F., and Adam, H. (2017). Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*. 57

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In Ferrari, V., Hebert, M., Sminchisescu, C., and Weiss, Y., editors, *Computer Vision ECCV 2018. Lecture Notes in Computer Science, vol 11211.*, pages 833–851. Springer, Cham. 35, 73, 82, 112

Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., Chen, K., et al. (2015). Xgboost: extreme gradient boosting. *R package version 0.4-2*, 1(4):1–4. 8

Chen, X., Xiang, S., Liu, C. L., and Pan, C. H. (2014). Vehicle detection in satellite images by hybrid deep convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 11(10):1797–1801. 52

Chen, Z., Wang, C., Wen, C., Teng, X., Chen, Y., Guan, H., Luo, H., Cao, L., and Li, J. (2016). Vehicle detection in high-resolution aerial images via sparse representation and superpixels. *IEEE Transactions on Geoscience and Remote Sensing*, 54(1):103–116. 48

Cheng, B., Collins, M. D., Zhu, Y., Liu, T., Huang, T. S., Adam, H., and Chen, L.-C. (2020a). Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12475–12485. 106

Cheng, G., Xie, X., Han, J., Guo, L., and Xia, G.-S. (2020b). Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:3735–3756. 14

Cheng, H. Y., Weng, C. C., and Chen, Y. Y. (2012). Vehicle detection in aerial surveillance using dynamic bayesian networks. *IEEE Transactions on Image Processing*, 21(4):2152–2159. 48

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. volume 29, pages 3213–3223, Las Vegas, NV, USA. IEEE. 2, 9, 22, 107

Costa, d. M. V. C. V., Carvalho, d. O. L. F., Orlandi, A. G., Hirata, I., Albuquerque, D. A. O., Silva, F. V. e., Guimarães, R. F., Gomes, R. A. T., and Júnior, O. A. d. C. (2021). Remote sensing for monitoring photovoltaic solar plants in brazil using deep semantic segmentation. *Energies*, 14(10):2960. 25, 30, 58, 69, 94

da Costa, L. B., de Carvalho, O. L. F., de Albuquerque, A. O., Gomes, R. A. T., Guimarães, R. F., and de Carvalho Júnior, O. A. (2021a). Deep semantic segmentation for detecting eucalyptus planted forests in the brazilian territory using sentinel-2 imagery. *Geocarto International*, (just-accepted):1–12. 94

da Costa, L. B., de Carvalho, O. L. F., de Albuquerque, A. O., Gomes, R. A. T., Guimarães, R. F., and de Carvalho Júnior, O. A. (2021b). Deep semantic segmentation for detecting eucalyptus planted forests in the brazilian territory using sentinel-2 imagery. *Geocarto International*, 0(0):1–13. 19, 25, 35, 43, 58, 69

Dacey, S., Song, L., and Pang, S. (2013). An intelligent agent based land encroachment detection approach. pages 585–592. 12

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. volume 1, pages 886–893, San Diego, CA, USA. IEEE. 49

Das, S., Mirnalinee, T. T., and Varghese, K. (2011). Use of salient features for the design of a multistage framework to extract roads from high-resolution multispectral satellite images. *IEEE Transactions on Geoscience and Remote Sensing*, 49(10):3906–3931. 91

de Albuquerque, A. O., de Carvalho, O. L. F., e Silva, C. R., de Bem, P. P., Gomes, R. A. T., Borges, D. L., Guimarães, R. F., Pimentel, C. M. M., and de Carvalho Júnior, O. A. (2021a). Instance segmentation of center pivot irrigation systems using multi-temporal sentinel-1 sar images. *Remote Sensing Applications: Society and Environment*, 23:100537. 119

de Albuquerque, A. O., de Carvalho, O. L. F., e Silva, C. R., Luiz, A. S., Pablo, P., Gomes, R. A. T., Guimarães, R. F., and de Carvalho Júnior, O. A. (2021b). Dealing with clouds and seasonal changes for center pivot irrigation systems detection using instance segmentation in sentinel-2 time series. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:8447–8457. 119

de Albuquerque, A. O., de Carvalho Júnior, O. A., Carvalho, O. L. F. d., de Bem, P. P., Ferreira, P. H. G., de Moura, R. d. S., Silva, C. R., Trancoso Gomes, R. A., and Fontes Guimarães, R. (2020a). Deep semantic segmentation of center pivot irrigation systems from remotely sensed data. *Remote Sensing*, 12(13):2159. 94, 97, 100

de Albuquerque, A. O., de Carvalho Júnior, O. A., Carvalho, d. O. L. F., de Bem, P. P., Ferreira, P. H. G., de Moura, R. d. S., Silva, C. R., Trancoso Gomes, R. A., and Fontes Guimarães, R. (2020b). Deep semantic segmentation of center pivot irrigation systems from remotely sensed data. *Remote Sensing*, 12(13):2159. 19, 25, 35, 43, 58, 69

de Bem, P. P., de Carvalho Júnior, O. A., de Carvalho, O. L. F., Gomes, R. A. T., Guimarães, R. F., and Pimentel, C. M. M. (2021). Irrigated rice crop identification in southern brazil using convolutional neural networks and sentinel-1 time series. *Remote Sensing Applications: Society and Environment*, 24:100627. 119

de Carvalho, O. L. F., de Carvalho Júnior, O. A., de Albuquerque, A. O., Santana, N. C., Guimarães, R. F., Gomes, R. A. T., and Borges, D. L. (2022a). Bounding box-free instance segmentation using semi-supervised iterative learning for vehicle detection. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:3403–3420. 119

de Carvalho, O. L. F., de Carvalho Júnior, O. A., Silva, C. R. e., de Albuquerque, A. O., Santana, N. C., Borges, D. L., Gomes, R. A. T., and Guimarães, R. F. (2022b). Panoptic segmentation meets remote sensing. *Remote Sensing*, 14(4):965. 69

de Carvalho, O. L. F., de Carvalho Júnior, O. A., Silva, C. R. e., de Albuquerque, A. O., Santana, N. C., Borges, D. L., Gomes, R. A. T., and Guimarães, R. F. (2022c). Panoptic segmentation meets remote sensing. *Remote Sensing*, 14(4):965. 106, 107, 109, 111, 112, 120

De Carvalho, O. L. F., De Carvalho Júnior, O. A., De Albuquerque, A. O., Santana, N. C., and Borges, D. L. (2022). Rethinking panoptic segmentation in remote sensing: A hybrid approach using semantic segmentation and non-learning methods. *IEEE Geoscience and Remote Sensing Letters*, pages 1–1. 106, 119

de Carvalho, O. L. F., de Carvalho Júnior, O. A., de Albuquerque, A. O., Santana, N. C., Borges, D. L., Gomes, R. A. T., and Guimarães, R. F. (2021a). Bounding box-free instance segmentation using semi-supervised learning for generating a city-scale vehicle dataset. 11, 31, 33, 35, 44, 97, 101

de Carvalho, O. L. F., de Carvalho Júnior, O. A., e Silva, C. R., de Albuquerque, A. O., Santana, N. C., Borges, D. L., Gomes, R. A. T., and Guimarães, R. F. (2021b). Panoptic segmentation meets remote sensing. 96, 97

de Carvalho, O. L. F., de Moura, R. d. S., de Albuquerque, A. O., de Bem, P. P., Pereira, R. d. C., Weigang, L., Borges, D. L., Guimarães, R. F., Gomes, R. A. T., and de Carvalho Júnior, O. A. (2021c). Instance segmentation for governmental inspection of small touristic infrastructure in beach zones using multispectral high-resolution worldview-3 imagery. *ISPRS International Journal of Geo-Information*, 10(12):813. 2, 68, 94

de Carvalho, O. L. F., de Moura, R. d. S., de Albuquerque, A. O., de Bem, P. P., Pereira, R. d. C., Weigang, L., Borges, D. L., Guimarães, R. F., Gomes, R. A. T., and de Carvalho Júnior, O. A. (2021d). Instance segmentation for governmental inspection of small touristic infrastructure in beach zones using multispectral high-resolution worldview-3 imagery. *ISPRS International Journal of Geo-Information*, 10(12):813. 107, 120

de Carvalho, O. L. F., Júnior, O. A. d. C., de Albuquerque, A. O., Santana, N. C., Borges, D. L., Gomes, R. A. T., and Guimarães, R. F. (2021e). Bounding box-free instance segmentation using semi-supervised learning for generating a city-scale vehicle dataset. *arXiv preprint arXiv:2111.12122*. 76

DeFries, R., Hansen, A., Turner, B. L., Reid, R., and Liu, J. (2007). Land use change around protected areas: management to balance human needs and ecological function. *Ecological Applications*, 17(4):1031–1038. 12

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. pages 248–255, Miami, FL, USA. IEEE. 73

Deng, Z., Sun, H., Zhou, S., Zhao, J., and Zou, H. (2017). Toward fast and accurate vehicle detection in aerial images using coupled region-based convolutional neural networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(8):3652–3664. 52

Dhillon, A. and Verma, G. K. (2020). Convolutional neural network: a review of models, methodologies and applications to object detection. *Progress in Artificial Intelligence*, 9(2):85–112. 1

Drew, P. J. and Monson, J. R. (2000). Artificial neural networks. *Surgery*, 127(1):3–11. 8

Drouyer, S. (2020). Vehsat: a large-scale dataset for vehicle detection in satellite images. pages 268–271. IEEE. 69, 91

Eikvil, L., Aurdal, L., and Koren, H. (2009). Classification-based vehicle detection in high-resolution satellite images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64(1):65–72. 48

El Mahrad, B., Newton, A., Icely, J., Kacimi, I., Abalansa, S., and Snoussi, M. (2020). Contribution of remote sensing technologies to a holistic coastal and marine environmental management framework: A review. *Remote Sensing*, 12(14):2313. 13

El Naqa, I. and Murphy, M. J. (2015). What is machine learning? In *machine learning in radiation oncology*, pages 3–11. Springer. 5

Everingham, M., Eslami, S. M. A., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136. 9

Fachrie, M. (2020). A simple vehicle counting system using deep learning with yolov3 model. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 4(3):462–468. 46

Feng, D., Haase-Schutz, C., Rosenbaum, L., Hertlein, H., Glaser, C., Timm, F., Wiesbeck, W., and Dietmayer, K. (2021). Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1341–1360. 45

Ferraz de Andrade Santos, J. A., de Jong, P., Alves da Costa, C., and Torres, E. A. (2020). Combining wind and solar energy sources: Potential for hybrid power generation in brazil. *Utilities Policy*, 67(June). 29

Filgueiras, A. and Thelma Maria, V. T. M. (2003). Wind energy in brazil - present and future. *Renewable and Sustainable Energy Reviews*, 7(5):439–451. 29

Gao, S. H., Cheng, M. M., Zhao, K., Zhang, X. Y., Yang, M. H., and Torr, P. (2021). Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*, 43(2):652–662. 71, 84, 85, 107, 112

Gao, Z., Ji, H., Mei, T., Ramesh, B., and Liu, X. (2019). Eovnet: Earth-observation image-based vehicle detection network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(9):3552–3561. 52

Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., and Garcia-Rodriguez, J. (2017). A review on deep learning techniques applied to semantic segmentation. *arXiv*, pages 1–23. 33

Garnot, V. S. F. and Landrieu, L. (2021). Panoptic segmentation of satellite image time series with convolutional temporal attention networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4872–4881. 72, 94, 106

Gauthier, C., Lin, Z., Peter, B. G., and Moran, E. F. (2019). Hydroelectric infrastructure and potential groundwater contamination in the brazilian amazon: Altamira and the belo monte dam. *Professional Geographer*, 71(2):292–300. 28

Gauthier, C. and Moran, E. F. (2018). Public policy implementation and basic sanitation issues associated with hydroelectric projects in the brazilian amazon: Altamira and the belo monte dam. *Geoforum*, 97(February):10–21. 28

Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics: The kitti dataset. *International Journal of Robotics Research*, 32(11):1231–1237. 9

Ghassemian, H. (2016). A review of remote sensing image fusion methods. *Information Fusion*, 32:75–89. 16

Girshick, R. (2015). Fast r-cnn. volume 2015 Inter, pages 1440–1448, Santiago, Chile. IEEE. 18, 51, 82

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. volume 1, pages 580–587, Columbus, OH, USA. IEEE. 18, 51

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2016). Region-based convolutional networks for accurate object detection and segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(1):142–158. 55

Gladstone, W., Curley, B., and Shokri, M. R. (2013). Environmental impacts of tourism in the gulf and the red sea. *Marine Pollution Bulletin*, 72(2):375–388. 13

Gleason, J., Nefian, V. A., Bouyssounousse, X., Fong, T., and Bebis, G. (2011). Vehicle detection from aerial imagery. pages 2065–2070. IEEE. 50

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press. 4

Grabner, H., Nguyen, T. T., Gruber, B., and Bischof, H. (2008). On-line boosting-based car detection from aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 63(3):382–396. 50

Griffiths, D. and Boehm, J. (2019). Improving public data for building segmentation from convolutional neural networks (cnns) for fused airborne lidar and image data using active contours. *ISPRS Journal of Photogrammetry and Remote Sensing*, 154(May):70–83. 73

Guirado, E., Tabik, S., Alcaraz-Segura, D., Cabello, J., and Herrera, F. (2017). Deep-learning versus obia for scattered shrub detection with google earth imagery: Ziziphus lotus as case study. *Remote Sensing*, 9(12):1220. 14

Guo, H., He, G., Jiang, W., Yin, R., Yan, L., and Leng, W. (2020a). A multi-scale water extraction convolutional neural network (mwen) method for gaofen-1 remote sensing images. *ISPRS International Journal of Geo-Information*, 9(4):189. 73

Guo, Q. and Wang, Z. (2020). A self-supervised learning framework for road centerline extraction from high-resolution remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:4451–4461. 73

Guo, Y., Liu, Y., Georgiou, T., and Lew, M. S. (2018). A review of semantic segmentation using deep neural networks. *International Journal of Multimedia Information Retrieval*, 7(2):87–93. 47

Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., and Lew, M. S. (2016). Deep learning for visual understanding: A review. *Neurocomputing*, 187:27–48. 33

Guo, Y., Xu, Y., and Li, S. (2020b). Dense construction vehicle detection based on orientation-aware feature fusion convolutional neural network. *Automation in Construction*, 112(August 2019):103124. 52

Hafiz, A. M. and Bhat, G. M. (2020). A survey on instance segmentation: state of the art. *International Journal of Multimedia Information Retrieval*, 9(3):171–189. 47

Ham, S. W., Park, H. C., Kim, E. J., Kho, S. Y., and Kim, D. K. (2020). Investigating the influential factors for practical application of multi-class vehicle detection for images from unmanned aerial vehicle using deep learning models. *Transportation Research Record*, 2674(12):553–567. 52

Han, J., Zhang, D., Cheng, G., Guo, L., and Ren, J. (2015). Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning. *IEEE Transactions on Geoscience and Remote Sensing*, 53(6):3325–3337. 49

Hao, Z., Lin, L., Post, C. J., Mikhailova, E. A., Li, M., Chen, Y., Yu, K., and Liu, J. (2021). Automated tree-crown and height detection in a young forest plantation using mask region-based convolutional neural network (mask r-cnn). *ISPRS Journal of Photogrammetry and Remote Sensing*, 178(May):112–123. 24

He, H., Yang, D., Wang, S., Wang, S., and Li, Y. (2019a). Road extraction by using atrous spatial pyramid pooling integrated encoder-decoder network and structural similarity loss. *Remote Sensing*, 11(9):1015. 73

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969. 16, 96

He, K., Gkioxari, G., Dollar, P., and Girshick, R. (2020). Mask r-cnn. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):386–397. 10, 33, 51, 55, 57, 82, 84, 107, 111, 112

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778. 18, 35, 57, 112

He, Y., Ma, W., Ma, Z., Fu, W., Chen, C., Yang, C.-F., and Liu, Z. (2019b). Using unmanned aerial vehicle remote sensing and a monitoring information system to enhance the management of unauthorized structures. *Applied Sciences*, 9(22):4954. 13

Heidler, K., Mou, L., Baumhoer, C., Dietz, A., and Zhu, X. X. (2021). Hed-unet: Combined segmentation and edge detection for monitoring the antarctic coastline. *IEEE Transactions on Geoscience and Remote Sensing*, pages 1–14. 97

Hinz, S. (2003). Detection and counting of cars in aerial images. volume 2, pages III–997–1000, Barcelona, Spain. IEEE. 48

Hoeser, T., Bachofer, F., and Kuenzer, C. (2020). Object detection and image segmentation with deep learning on earth observation data: A review-part ii: Applications. *Remote Sensing*, 12(18). 10, 14

Holt, A. C., Seto, E. Y., Rivard, T., and Gong, P. (2009). Object-based detection and classification of vehicles from high-resolution aerial photography. *Photogrammetric Engineering and Remote Sensing*, 75(7):871–880. 48

Hossain, M. D. and Chen, D. (2019). Segmentation for object-based image analysis (obia): A review of algorithms and challenges from remote sensing perspective. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150(November 2018):115–134. 48

Hou, X., Ao, W., Song, Q., Lai, J., Wang, H., and Xu, F. (2020). Fusar-ship: building a high-resolution sar-ais matchup dataset of gaofen-3 for ship detection and recognition. *Science China Information Sciences*, 63(4):140303. 91

Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., et al. (2019). Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1314–1324. 71

Hua, X., Wang, X., Rui, T., Shao, F., and Wang, D. (2021). Cascaded panoptic segmentation method for high resolution remote sensing image. *Applied Soft Computing*, 109:107515. 72, 106

Huang, H., Lan, Y., Yang, A., Zhang, Y., Wen, S., and Deng, J. (2020). Deep learning versus object-based image analysis (obia) in weed mapping of uav imagery. *International Journal of Remote Sensing*, 41(9):3446–3479. 14

Huang, L., Liu, B., Li, B., Guo, W., Yu, W., Zhang, Z., and Yu, W. (2018). Opensarship: A dataset dedicated to sentinel-1 ship interpretation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(1):195–208. 91

Huang, Z., Huang, L., Gong, Y., Huang, C., and Wang, X. (2019). Mask scoring r-cnn. pages 6402–6411, Long Beach, CA, USA, USA. IEEE. 84, 112

Hunt., J. D., Stilpen, D., and de Freitas, M. A. V. (2018). A review of the causes, impacts and solutions for electricity supply crises in brazil. *Renewable and Sustainable Energy Reviews*, 88(January 2017):208–222. 29

Ibarra-Marinas, D., Belmonte-Serrato, F., Ballesteros-Pelegrín, G., and García-Marín, R. (2021). Evolution of the beaches in the regional park of salinas and arenales of san pedro del pinatar (southeast of spain) (18992019). *ISPRS International Journal of Geo-Information*, 10(4):200. 13

Jakovljevic, G., Govedarica, M., and Alvarez-Taboada, F. (2020). A deep learning model for automatic plastic mapping using unmanned aerial vehicle (uav) data. *Remote Sensing*, 12(9):1515. 73

Janai, J., Güney, F., Behl, A., and Geiger, A. (2020). Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trendső in Computer Graphics and Vision*, 12(13):1–308. 45

Janocha, K. and Czarnecki, W. M. (2017). On loss functions for deep neural networks in classification. *arXiv preprint arXiv:1702.05659*. 7

Javadi, S., Dahl, M., and Pettersson, M. I. (2021). Vehicle detection in aerial images based on 3d depth maps and deep neural networks. *IEEE Access*, 9:8381–8391. 52

Ji, H., Gao, Z., Mei, T., and Li, Y. (2019a). Improved faster r-cnn with multiscale feature fusion and homography augmentation for vehicle detection in remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, 16(11):1–5. 45, 52

Ji, S., Wei, S., and Lu, M. (2019b). Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Transactions on Geoscience and Remote Sensing*, 57(1):574–586. 91

Jiang, S., Yao, W., Wong, M. S., Li, G., Hong, Z., Kuc, T. Y., and Tong, X. (2020). An optimized deep neural network detecting small and narrow rectangular objects in google earth images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:1068–1081. 52

Jiang, X., Lu, D., Moran, E., Calvi, M. F., Dutra, L. V., and Li, G. (2018). Examining impacts of the belo monte hydroelectric dam construction on land-cover changes using multitemporal landsat imagery. *Applied Geography*, 97(May):35–47. 28

Jie, Y., Ji, X., Yue, A., Chen, J., Deng, Y., Chen, J., and Zhang, Y. (2020). Combined multi-layer feature fusion and edge detection method for distributed photovoltaic power station identification. *Energies*, 13(24):6742. 30

Johansen, K., Duan, Q., Tu, Y.-H., Searle, C., Wu, D., Phinn, S., Robson, A., and McCabe, M. F. (2020). Mapping the condition of macadamia tree crops using multi-spectral uav and worldview-3 imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 165:28–40. 15

Kaur, P., Krishan, K., Sharma, S. K., and Kanchan, T. (2020). Facial-recognition algorithms: A literature review. *Medicine, Science and the Law*, 60(2):131–139. 1, 14

Kembhavi, A., Harwood, D., and Davis, L. S. (2011). Vehicle detection using partial least squares. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(6):1250–1265. 50

Kestur, R., Farooq, S., Abdal, R., Mehraj, E., Narasipura, O., and Mudigere, M. (2018). Ufcn: a fully convolutional neural network for road extraction in rgb imagery acquired by remote sensing from an unmanned aerial vehicle. *Journal of Applied Remote Sensing*, 12(01):1. 73

Khelifi, L. and Mignotte, M. (2020). Deep learning for change detection in remote sensing images: Comprehensive review and meta-analysis. *IEEE Access*, 8:126385–126400. 14

Khoshboresh-Masouleh, M. and Shah-Hosseini, R. (2021). Building panoptic change segmentation with the use of uncertainty estimation in squeeze-and-attention cnn and remote sensing observations. *International Journal of Remote Sensing*, 42(20):7798–7820. 72, 94, 106

Kim, D., Woo, S., Lee, J.-Y., and Kweon, I. S. (2020). Video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9859–9868. 106

Kirillov, A., He, K., Girshick, R., Rother, C., and Dollár, P. (2019). Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9404–9413. 10, 69, 71, 82, 85, 96, 100, 107, 111, 112

Kisantal, M., Wojna, Z., Murawski, J., Naruniec, J., and Cho, K. (2019). Augmentation for small object detection. *arXiv preprint arXiv:1902.07296*. 26, 120

Koga, Y., Miyazaki, H., and Shibasaki, R. (2018). A cnn-based method of vehicle detection from aerial images using hard example mining. *Remote Sensing*, 10(1). 52

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90. 49

Laben, C. A. and Brower, V. B. (2000). Process for enhancing the spatial resolution of multispectral imagery using pan-sharpening. US Patent 6,011,875. 15

Leberl, F., Bischof, H., Grabner, H., and Kluckner, S. (2007). Recognizing cars in aerial imagery to improve orthophotos. *GIS: Proceedings of the ACM International Symposium on Advances in Geographic Information Systems*, (May 2014):2–10. 50

Lecun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444. 1, 5, 14, 30

Lee, Y. and Park, J. (2020). Centermask: Real-time anchor-free instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13906–13915. 11

Leitloff, J., Rosenbaum, D., Kurz, F., Meynberg, O., and Reinartz, P. (2014). An operational system for estimating road traffic information from aerial images. *Remote Sensing*, 6(11):11315–11341. 49, 50

Li, J., Huang, X., and Gong, J. (2019a). Deep neural network for remote-sensing image interpretation: status and perspectives. *National Science Review*, 6(6):1082–1086. 14

Li, J., Liang, X., Wei, Y., Xu, T., Feng, J., and Yan, S. (2017). Perceptual generative adversarial networks for small object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1222–1230. 15, 25

Li, J., Meng, L., Yang, B., Tao, C., Li, L., and Zhang, W. (2021). Labelrs: An automated toolbox to make deep learning samples from remote sensing images. *Remote Sensing*, 13(11):2064. 77, 91, 107

Li, Q., Mou, L., Xu, Q., Zhang, Y., and Zhu, X. X. (2019b). R3-net: A deep network for multioriented vehicle detection in aerial images and videos. *IEEE Transactions on Geoscience and Remote Sensing*, 57(7):5028–5042. 52

Li, X. and Chen, D. (2021). A survey on deep learning-based panoptic segmentation. *Digital Signal Processing*, page 103283. 23

Li, X., Li, X., and Pan, H. (2020a). Multi-scale vehicle detection in high-resolution aerial images with context information. *IEEE Access*, 8:208643–208657. 14, 48, 52

Li, Y., Zhang, H., Xue, X., Jiang, Y., and Shen, Q. (2018). Deep learning for remote sensing image classification: A survey. *WIREs Data Mining and Knowledge Discovery*, 8(6). 14

Li, Z., Chen, G., and Zhang, T. (2020b). A cnn-transformer hybrid approach for crop classification using multitemporal multisensor images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:847–858. 119

Lian, R. and Huang, L. (2020). Deepwindow: Sliding window based on deep learning for road extraction from remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:1905–1916. 73

Liang, P., Teodoro, G., Ling, H., Blasch, E., Chen, G., and Bai, L. (2012). Multiple kernel learning for vehicle detection in wide area motion imagery. Number 1629, pages 1629–1636, Singapore. IEEE. 50

Lima, M. A., Mendes, L. F., Mothé, G. A., Linhares, F. G., de Castro, M. P., da Silva, M. G., and Sthel, M. S. (2020). Renewable energy in reducing greenhouse gas emissions: Reaching the goals of the paris agreement in brazil. *Environmental Development*, 33(November 2018):100504. 28

Lin, H.-Y., Tu, K.-C., and Li, C.-Y. (2020). Vaid: An aerial image dataset for vehicle detection and classification. *IEEE Access*, 8:212209–212219. 69, 70, 91

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125. 35, 57, 82, 112

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer. 2, 9, 15, 22, 31, 33, 47, 55, 71, 84, 96, 107, 112

Lin, Y., Zhang, H., Li, G., Wang, T., Wan, L., and Lin, H. (2019). Improving impervious surface extraction with shadow-based sparse representation from optical, sar, and lidar data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2417–2428. 91, 92

Lira, C. and Taborda, R. (2014). Advances in applied remote sensing to coastal environments using free satellite imagery. In Finkl, C. and Makowski, C., editors, *Remote Sensing and Modeling*, pages 77–102. Springer, Cham. 13

Liu, C., Ding, Y., Zhu, M., Xiu, J., Li, M., and Li, Q. (2019a). Vehicle detection in aerial images using a fast oriented region search and the vector of locally aggregated descriptors. *Sensors*, 19(15):3294. 50

Liu, C., Ke, W., Qin, F., and Ye, Q. (2018a). Linear span network for object skeleton detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 133–148. 72

Liu, K. and Mattyus, G. (2015). Fast multiclass vehicle detection on aerial images. *IEEE Geoscience and Remote Sensing Letters*, 12(9):1938–1942. 50

Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., and Pietikäinen, M. (2020). Deep learning for generic object detection: A survey. *International journal of computer vision*, 128(2):261–318. 14

Liu, S., Ding, W., Liu, C., Liu, Y., Wang, Y., and Li, H. (2018b). Ern: Edge loss reinforced semantic segmentation network for remote sensing images. *Remote Sensing*, 10(9):1339. 92

Liu, S., Wang, Y., Yang, X., Lei, B., Liu, L., Li, S. X., Ni, D., and Wang, T. (2019b). Deep learning in medical ultrasound analysis: A review. *Engineering*, 5(2):261–275. 1, 14

Liu, T., Abd-Elrahman, A., Morton, J., and Wilhelm, V. L. (2018c). Comparing fully convolutional networks, random forest, support vector machine, and patch-based deep convolutional neural networks for object-based wetland mapping using images from small unmanned aircraft system. *GIScience Remote Sensing*, 55(2):243–264. 14

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd: Single shot multibox detector. In *Computer Vision ECCV 2016. Lecture Notes in Computer Science*, volume 9905, pages 21–37. Springer, Cham. 51

Liu, X., Yang, T., and Li, J. (2018d). Real-time ground vehicle detection in aerial infrared imagery based on convolutional neural network. *Electronics (Switzerland)*, 7(6):1–19. 52

Llausàs, A., Hof, A., Wolf, N., Saurí, D., and Siegmund, A. (2019). Applicability of cadastral data to support the estimation of water use in private swimming pools. *Environment and Planning B: Urban Analytics and City Science*, 46(6):1165–1181. 14

Lv, Y., Zhang, C., Yun, W., Gao, L., Wang, H., Ma, J., Li, H., and Zhu, D. (2020). The delineation and grading of actual crop production units in modern smallholder areas using rs data and mask r-cnn. *Remote Sensing*, 12(7):1074. 23

Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G., and Johnson, B. A. (2019). Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 152(November 2018):166–177. 2, 10, 14

Madhogaria, S., Baggenstoss, P., Schikora, M., Koch, W., and Cremers, D. (2015). Car detection by fusion of hog and causal mrf. *IEEE Transactions on Aerospace and Electronic Systems*, 51(1):575–590. 50

Maggiori, E., Tarabalka, Y., Charpiat, G., and Alliez, P. (2017). Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark. pages 3226–3229, Fort Worth, TX, USA. IEEE. 91

Mandal, M., Shah, M., Meena, P., Devi, S., and Vipparthi, S. K. (2020). Avdnet: A small-sized vehicle detection network for aerial visual data. *IEEE Geoscience and Remote Sensing Letters*, 17(3):494–498. 52

Martin, C., Parkes, S., Zhang, Q., Zhang, X., McCabe, M. F., and Duarte, C. M. (2018). Use of unmanned aerial vehicles for efficient beach litter monitoring. *Marine pollution bulletin*, 131:662–673. 12

Mayer, A., Castro-Diaz, L., Lopez, M. C., Leturcq, G., and Moran, E. F. (2021). Is hydropower worth it? exploring amazonian resettlement, human development and environmental costs with the belo monte project in brazil. *Energy Research and Social Science*, 78(May):102129. 28, 29

McCarthy, M. J., Colna, K. E., El-Mezayen, M. M., Laureano-Rosario, A. E., Méndez-Lázaro, P., Otis, D. B., Toro-Farmer, G., Vega-Rodriguez, M., and Muller-Karger, F. E. (2017). Satellite remote sensing for coastal management: A review of successful applications. *Environmental Management*, 60(2):323–339. 13

Melo, L. B., Estanislau, F. B., Costa, A. L., and Fortini, (2019). Impacts of the hydrological potential change on the energy matrix of the brazilian state of minas gerais: A case study. *Renewable and Sustainable Energy Reviews*, 110(December 2018):415–422. 29

Mendes, L. F. R. and Sthel, M. S. (2017). Thermoelectric power plant for compensation of hydrological cycle change: Environmental impacts in brazil. *Case Studies in the Environment*, 1(1):1–7. 29

Mendes, L. F. R. and Sthel, M. S. (2018). Analysis of the hydrological cycle and its impacts on the sustainability of the electric matrix in the state of rio de janeiro/brazil. *Energy Strategy Reviews*, 22(July):119–126. 29

Milosavljevic, A. (2020). Automated processing of remote sensing imagery using deep semantic segmentation: a building footprint extraction case. *ISPRS International Journal of Geo-Information*, 9(8):486. 73

Mohan, R. and Valada, A. (2021a). Efficientps: Efficient panoptic segmentation. *International Journal of Computer Vision*, 129(5):1551–1579. 71, 85, 107, 112

Mohan, R. and Valada, A. (2021b). Efficientps: Efficient panoptic segmentation. *International Journal of Computer Vision*, 129(5):1551–1579. 96

Mokhtarzade, M. and Zoej, M. J. V. (2007). Road detection from high-resolution satellite images using artificial neural networks. *International Journal of Applied Earth Observation and Geoinformation*, 9(1):32–40. 73

Momeni, R., Aplin, P., and Boyd, D. (2016). Mapping complex urban land cover from spaceborne imagery: The influence of spatial resolution, spectral band set and classification approach. *Remote Sensing*, 8(2):88. 13

Moranduzzo, T. and Melgani, F. (2014a). Automatic car counting method for unmanned aerial vehicle images. *IEEE Transactions on Geoscience and Remote Sensing*, 52(3):1635–1647. 49, 50

Moranduzzo, T. and Melgani, F. (2014b). Detecting cars in uav images with a catalog-based approach. *IEEE Transactions on Geoscience and Remote Sensing*, 52(10):6356–6367. 49, 50

Mou, L. and Zhu, X. X. (2018). Vehicle instance segmentation from aerial image and video using a multitask learning residual fully convolutional network. *IEEE Transactions on Geoscience and Remote Sensing*, 56(11):6699–6711. 33, 35, 44, 47, 51, 55, 68, 73, 97, 101, 119

Murtagh, F. (1991). Multilayer perceptrons for classification and regression. *Neurocomputing*, 2(5-6):183–197. 5

Nassif, A. B., Shahin, I., Attili, I., Azzeh, M., and Shaalan, K. (2019). Speech recognition using deep neural networks: A systematic review. *IEEE Access*, 7:19143–19165. 1, 14

Neuhold, G., Ollmann, T., Bulo, S. R., and Kontschieder, P. (2017). The mapillary vistas dataset for semantic understanding of street scenes. volume 2017-Octob, pages 5000–5009. IEEE. 2, 9, 22, 107

Nguyen, T. T., Grabner, H., Bischof, H., and Gruber, B. (2007). On-line boosting for car detection from aerial images. *2007 IEEE International Conference on Research, Innovation and Vision for the Future, RIVF 2007*, (June 2014):87–95. 49, 50

Nogueira, K., Penatti, O. A., and dos Santos, J. A. (2017). Towards better exploiting convolutional neural networks for remote sensing scene classification. *Pattern Recognition*, 61:539–556. 1, 14, 30

Ophoff, T., Puttemans, S., Kalogirou, V., Robin, J.-P., and Goedemé, T. (2020). Vehicle and vessel detection on satellite imagery: A comparative study on single-shot detectors. *Remote Sensing*, 12(7):1217. 52

Orlandi, A. G., Farias, R. A. N., de Carvalho Junior, O. A., Guimarães, R., and Gomes, R. (2021). Controle gerencial na administração pública e transformação digital: sensoriamento remoto. *Cadernos Gestão Pública e Cidadania*, 26(83):1–24. 29

Ouellette, W. and Getinet, W. (2016). Remote sensing for marine spatial planning and integrated coastal areas management: Achievements, challenges, opportunities and future prospects. *Remote Sensing Applications: Society and Environment*, 4:138–157. 13

Paoletti, M., Haut, J., Plaza, J., and Plaza, A. (2019). Deep learning classifiers for hyperspectral imaging: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 158:279–317. 14

Papakonstantinou, A., Doukari, M., Stamatis, P., and Topouzelis, K. (2019). Coastal management using uas and high-resolution satellite images for touristic areas. *International Journal of Applied Geospatial Research*, 10(1):54–72. 14

Parikh, H., Patel, S., and Patel, V. (2020). Classification of sar and polsar images using deep learning: a review. *International Journal of Image and Data Fusion*, 11(1):1–32. 14

Parthasarathy, K. and Deka, P. C. (2019). Remote sensing and gis application in assessment of coastal vulnerability and shoreline changes: a review. *ISH Journal of Hydraulic Engineering*, pages 1–13. 13

Plakman, V., Rosier, J., and van Vliet, J. (2022). Solar park detection from publicly available satellite imagery. *GIScience and Remote Sensing*, 59(1):461–480. 30

Poompavai, V. and Ramalingam, M. (2013). Geospatial analysis for coastal risk assessment to cyclones. *Journal of the Indian Society of Remote Sensing*, 41(1):157–176. 13

Qiao, S., Zhu, Y., Adam, H., Yuille, A., and Chen, L.-C. (2021). Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3997–4008. 106

Qu, S., Wang, Y., Meng, G., and Pan, C. (2016). Vehicle detection in satellite images by incorporating objectness and convolutional neural network. *Journal of Industrial and Intelligent Information*, 4(2):158–162. 52

Rastogi, K., Bodani, P., and Sharma, S. A. (2020). Automatic building footprint extraction from very high-resolution imagery using deep learning techniques. *Geocarto International*, 0(0):1–13. 73

Razakarivony, S. and Jurie, F. (2016). Vehicle detection in aerial imagery: A small target detection benchmark. *Journal of Visual Communication and Image Representation*, 34:187–203. 50

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. pages 779–788, Las Vegas, NV, USA. IEEE. 51

Rego, E. E. and de Oliveira Ribeiro, C. (2018). Successful brazilian experience for promoting wind energy generation. *Electricity Journal*, 31(2):13–17. 43

Reichert, B. and Souza, A. M. (2021). Interrelationship simulations among brazilian electric matrix sources. *Electric Power Systems Research*, 193:107019. 29

Reksten, J. H. and Salberg, A. B. (2021). Estimating traffic in urban areas from very-high resolution aerial images. *International Journal of Remote Sensing*, 42(3):865–883. 51

Ren, S., He, K., Girshick, R., and Sun, J. (2017). Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149. 18, 51

Ren, Y., Zhu, C., and Xiao, S. (2018). Small object detection in optical remote sensing images via modified faster r-cnn. *Applied Sciences*, 8(5):813. 25

Rifat, S. and Liu, W. (2020). Measuring community disaster resilience in the conterminous coastal united states. *ISPRS International Journal of Geo-Information*, 9(8):469. 13

Ronneberger, O., Fischer, P., and Brox, T. (2015a). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer. 33, 98

Ronneberger, O., Fischer, P., and Brox, T. (2015b). U-net: Convolutional networks for biomedical image segmentation. In Navab, N., Hornegger, J., Wells, W., and Frangi, A., editors, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 9351, pages 234–241. Springer, Cham. 57, 82, 112

Runde, A., Hallwass, G., and Silvano, R. A. (2020). Fishers' knowledge indicates extensive socioecological impacts downstream of proposed dams in a tropical river. *One Earth*, 2(3):255–268. 28

Russell, B. C., Torralba, A., Murphy, K. P., and Freeman, W. T. (2008). Labelme: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1-3):157–173. 16, 76, 90

Sahana, M., Hong, H., Ahmed, R., Patel, P. P., Bhakat, P., and Sajjad, H. (2019). Assessing coastal island vulnerability in the sundarban biosphere reserve, india, using geospatial technology. *Environmental Earth Sciences*, 78(10):304. 13

Sakhare, V. K., Tewari, T., and Vyas, V. (2020). Review of vehicle detection systems in advanced driver assistant systems. *Archives of Computational Methods in Engineering*, 27(2):591–610. 45

Sampaio, P. G. V. and González, M. O. A. (2017). Photovoltaic solar energy: Conceptual framework. *Renewable and Sustainable Energy Reviews*, 74(June 2016):590–601. 29

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61:85–117. 1, 14

Sekachev, B., Nikita, M., and Andrey, Z. (2019). Computer vision annotation tool: A universal approach to data annotation. [Online; accessed 2021-10-30]. 16, 76

Senthilnath, J., Varia, N., Dokania, A., Anand, G., and Benediktsson, J. A. (2020). Deep tec: Deep transfer learning with ensemble classifier for road extraction from uav imagery. *Remote Sensing*, 12(2):245. 73

Serra-Gonçalves, C., Lavers, J. L., and Bond, A. L. (2019). Global review of beach debris monitoring and future recommendations. *Environmental science technology*, 53(21):12158–12167. 12

Serte, S., Serener, A., and AlTurjman, F. (2020). Deep learning in medical imaging: A brief review. *Transactions on Emerging Telecommunications Technologies*. 1, 14

Sevo, I. and Avramovic, A. (2016). Convolutional neural network based automatic object detection on aerial images. *IEEE Geoscience and Remote Sensing Letters*, 13(5):740–744. 49

Shao, W., Yang, W., Liu, G., and Liu, J. (2012). Car detection from high-resolution aerial imagery using multiple features. *International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 4379–4382. 50

Sharma, G., Merry, C. J., Goel, P., and McCord, M. (2006). Vehicle detection in 1m resolution satellite and airborne imagery. *International Journal of Remote Sensing*, 27(4):779–797. 48

Sharma, V. and Mir, R. N. (2020). A comprehensive and systematic look up into deep learning based object detection techniques: A review. *Computer Science Review*, 38:100301. 1, 14

Shelhamer, E., Long, J., and Darrell, T. (2017). Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):640–651. 18

Shen, G., Xu, B., Jin, Y., Chen, S., Zhang, W., Guo, J., Liu, H., Zhang, Y., and Yang, X. (2017). Monitoring wind farms occupying grasslands based on remote-sensing data from china's gf-2 hd satellitea case study of jiuquan city, gansu province, china. *Resources, Conservation and Recycling*, 121:128–136. 31

Shen, J., Liu, N., and Sun, H. (2019). Vehicle detection in aerial images based on hyper feature map in deep convolutional network. *KSII Transactions on Internet and Information Systems*, 13(4):479–491. 52

Shen, J., Liu, N., and Sun, H. (2021). Vehicle detection in aerial images based on lightweight deep convolutional network. *IET Image Processing*, 15(2):479–491. 48, 52

Shi, F., Zhang, T., and Zhang, T. (2021). Orientation-aware vehicle detection in aerial images via an anchor-free object detection approach. *IEEE Transactions on Geoscience and Remote Sensing*, 59(6):5221–5233. 48, 52

Signoroni, A., Savardi, M., Baronio, A., and Benini, S. (2019). Deep learning meets hyperspectral image analysis: A multidisciplinary review. *Journal of Imaging*, 5(5):52. 14

Simas, M. and Pacca, S. (2014). Assessing employment in renewable energy technologies: A case study for wind power in brazil. *Renewable and Sustainable Energy Reviews*, 31:83–90. 43

Singh, R. and Rani, R. (2020). Semantic segmentation using deep convolutional neural network: A review. *SSRN Electronic Journal*, pages 1–8. 10

Soloy, A., Turki, I., Fournier, M., Costa, S., Peuziat, B., and Lecoq, N. (2020). A deep learning-based method for quantifying and mapping the grain size on pebble beaches. *Remote Sensing*, 12(21):1–23. 23

Sommer, L., Schuchert, T., and Beyerer, J. (2019). Comprehensive analysis of deep learning-based vehicle detection in aerial images. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(9):2733–2747. 52

Song, H., Liang, H., Li, H., Dai, Z., and Yun, X. (2019). Vision-based vehicle detection and counting system using deep learning in highway scenes. *European Transport Research Review*, 11(1). 46

Stuparu, D. G., Ciobanu, R. I., and Dobre, C. (2020). Vehicle detection in overhead satellite images using a one-stage object detection model. *Sensors (Switzerland)*, 20(22):1–18. 52

Sun, S., Luo, C., and Chen, J. (2017). A review of natural language processing techniques for opinion mining systems. *Information Fusion*, 36:10–25. 1, 14

Sun, S., Mu, L., Wang, L., Liu, P., Liu, X., and Zhang, Y. (2021). Semantic segmentation for buildings of large intra-class variation in remote sensing images with o-gan. *Remote Sensing*, 13(3):475. 73

Tan, M. and Le, Q. (2019a). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR. 35, 98

Tan, M. and Le, V. Q. (2019b). Efficientnet: Rethinking model scaling for convolutional neural networks. pages 6105–6114, Long Beach, California, USA. 57, 112

Tan, Q., Ling, J., Hu, J., Qin, X., and Hu, J. (2020). Vehicle detection in high resolution satellite remote sensing images based on deep learning. *IEEE Access*, 8:153394–153402. 52

Tang, T., Zhou, S., Deng, Z., Zou, H., and Lei, L. (2017). Vehicle detection in aerial images based on region convolutional neural networks and hard negative example mining. *Sensors*, 17(2):336. 52

Tao, C., Mi, L., Li, Y., Qi, J., Xiao, Y., and Zhang, J. (2019). Scene context-driven vehicle detection in high-resolution aerial images. *IEEE Transactions on Geoscience and Remote Sensing*, 57(10):7339–7351. 51

Tayara, H., Gil Soo, K., and Chong, K. T. (2018). Vehicle detection and counting in high-resolution aerial images using convolutional regression neural network. *IEEE Access*, 6:2220–2230. 47, 51, 68

Tollefson, J. (2020). Brazil ratification pushes paris climate deal one step closer. *Nature*, (September):1–2. 28

Tong, K., Wu, Y., and Zhou, F. (2020). Recent advances in small object detection based on deep learning: A review. *Image and Vision Computing*, 97:103910. 15, 17, 25, 31, 68, 107, 120

Torralba, A., Russell, B. C., and Yuen, J. (2010). Labelme: Online image annotation and applications. *Proceedings of the IEEE*, 98(8):1467–1484. 16, 76

Tsoumakas, G. and Katakis, I. (2007). Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)*, 3(3):1–13. 9

Tuermer, S., Kurz, F., Reinartz, P., and Stilla, U. (2013). Airborne vehicle detection in dense urban areas using hog features and disparity maps. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 6(6):2327–2337. 49, 50

Vali, A., Comai, S., and Matteucci, M. (2020). Deep learning for land use and land cover classification based on hyperspectral and multispectral earth observation data: A review. *Remote Sensing*, 12(15):2495. 14

van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., and Yu, T. (2014). scikit-image: image processing in python. *PeerJ*, 2:e453. 56

Van Etten, A. (2018). You only look twice: Rapid multi-scale object detection in satellite imagery. 51

Van Etten, A., Lindenbaum, D., and Bacastow, T. M. (2018). Spacenet: A remote sensing dataset and challenge series. *arXiv preprint*, page arXiv:1807.01232. 73, 91

Varol, B., Ylmaz, E. Maktav, D., Bayburt, S., and Gürdal, S. (2019). Detection of illegal constructions in urban cities: comparing lidar data and stereo kompsat-3 images with development plans. *European Journal of Remote Sensing*, 52(1):335–344. 13

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. volume 1, pages I–511–I–518, Kauai, HI, USA. IEEE Comput. Soc. 49

Voulodimos, A., Doulamis, N., Doulamis, A., and Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018:1–13. 1, 10, 14

Vrabel, J. C., Stensaas, G. L., Anderson, C., Christopherson, J., Kim, M., Park, S., and Cantrell, S. (2021). System characterization report on the china-brazil earth resources satellite-4a (cbers–4a). In Ramaseri Chandra, S. N., editor, *System characterization of Earth observation sensors*, page 35. U.S. Geological Survey Open-File Report 20211030. 31

Wang, B. and Gu, Y. (2020). An improved fbpn-based detection network for vehicles in aerial images. *Sensors (Switzerland)*, 20(17):1–20. 52

Wang, H., Yu, Y., Cai, Y., Chen, X., Chen, L., and Liu, Q. (2019). A comparative study of state-of-the-art deep learning algorithms for vehicle detection. *IEEE Intelligent Transportation Systems Magazine*, 11(2):82–95. 45, 52

Wang, Q., Yan, L., Yuan, Q., and Ma, Z. (2017). An automatic shadow detection method for vhr remote sensing orthoimagery. *Remote Sensing*, 9(5):469. 92

Waqas Zamir, S., Arora, A., Gupta, A., Khan, S., Sun, G., Shahbaz Khan, F., Zhu, F., Shao, L., Xia, G.-S., and Bai, X. (2019). isaid: A large-scale dataset for instance segmentation in aerial images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 28–37. 91

Wei, S., Zeng, X., Qu, Q., Wang, M., Su, H., and Shi, J. (2020). Hrsid: A high-resolution sar images dataset for ship detection and instance segmentation. *IEEE Access*, 8:120234–120254. 91

Weng, L., Xu, Y., Xia, M., Zhang, Y., Liu, J., and Xu, Y. (2020). Water areas segmentation from remote sensing images using a separable residual segnet network. *ISPRS International Journal of Geo-Information*, 9(4):256. 73

Wu, Q., Luo, F., Wu, P., Wang, B., Yang, H., and Wu, Y. (2021). Automatic road extraction from high-resolution remote sensing images using a method based on densely connected spatial feature-enhanced pyramid. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:3–17. 23, 73

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., and Girshick, R. (2019). Detectron2. `https://github.com/facebookresearch/detectron2`. 2, 16, 18, 57, 71, 79, 96, 107, 111

Wurm, M., Stark, T., Zhu, X. X., Weigand, M., and Taubenböck, H. (2019). Semantic segmentation of slums in satellite images using transfer learning on fully convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150(February):59–69. 73

Xi, X., Yu, Z., Zhan, Z., Yin, Y., and Tian, C. (2019). Multi-task cost-sensitive-convolutional neural network for car detection. *IEEE Access*, 7:98061–98068. 46, 52

Xia, G.-s., Bai, X., Ding, J., Zhu, Z., Belongie, S., Luo, J., Datcu, M., Pelillo, M., and Zhang, L. (2018a). Dota: A large-scale dataset for object detection in aerial images. pages 3974–3983. IEEE. 69

Xia, G.-S., Bai, X., Ding, J., Zhu, Z., Belongie, S., Luo, J., Datcu, M., Pelillo, M., and Zhang, L. (2018b). Dota: A large-scale dataset for object detection in aerial images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3974–3983. 72

Xia, Z., Li, Y., Guo, X., and Chen, R. (2022). High-resolution mapping of water photovoltaic development in china through satellite imagery. *International Journal of Applied Earth Observation and Geoinformation*, 107(February):102707. 30

Xie, E., Sun, P., Song, X., Wang, W., Liu, X., Liang, D., Shen, C., and Luo, P. (2020). Polarmask: Single shot instance segmentation with polar representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12193–12202. 11

Xie, S., Girshick, R., Dollar, P., Tu, Z., and He, K. (2017). Aggregated residual transformations for deep neural networks. pages 5987–5995, Honolulu, HI, USA. IEEE. 18, 35, 57, 112

Xiong, Y., Liao, R., Zhao, H., Hu, R., Bai, M., Yumer, E., and Urtasun, R. (2019). Upsnet: A unified panoptic segmentation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8818–8826. 106

Xu, Y., Xie, Z., Feng, Y., and Chen, Z. (2018). Road extraction from high-resolution remote sensing imagery using deep learning. *Remote Sensing*, 10(9):1461. 73

Xu, Y., Yu, G., Wang, Y., Wu, X., and Ma, Y. (2016). A hybrid vehicle detection method based on viola-jones and hog + svm from uav images. *Sensors*, 16(8):1325. 50

Xu, Y., Yu, G., Wang, Y., Wu, X., and Ma, Y. (2017). Car detection from low-altitude uav imagery with the faster r-cnn. *Journal of Advanced Transportation*, 2017. 50, 52

Yakubovskiy, P. (2020). Segmentation models pytorch. *GitHub repository*. 57

Yang, M. Y., Liao, W., Li, X., Cao, Y., and Rosenhahn, B. (2019). Vehicle detection in aerial images. *Photogrammetric Engineering and Remote Sensing*, 85(4):297–304. 52

Yi, Y., Zhang, Z., Zhang, W., Zhang, C., Li, W., and Zhao, T. (2019). Semantic segmentation of urban buildings from vhr remote sensing imagery using a deep convolutional neural network. *Remote Sensing*, 11(15):1774. 73

Young, T., Hazarika, D., Poria, S., and Cambria, E. (2018). Recent trends in deep learning based natural language processing [review article]. *IEEE Computational Intelligence Magazine*, 13(3):55–75. 1, 14

Yu, X., Lou, B., Zhang, D., Winkel, D., Arrahmane, N., Diallo, M., Meng, T., Busch, H. v., Grimm, R., Kiefer, B., et al. (2020). Deep attentive panoptic model for prostate cancer detection using biparametric mri scans. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 594–604. Springer. 106

Yu, Y., Gu, T., Guan, H., Li, D., and Jin, S. (2019). Vehicle detection from high-resolution remote sensing imagery using convolutional capsule networks. *IEEE Geoscience and Remote Sensing Letters*, 16(12):1894–1898. 51

Yu, Y., Guan, H., and Ji, Z. (2015). Rotation-invariant object detection in high-resolution satellite imagery using superpixel-based deep hough forests. *IEEE Geoscience and Remote Sensing Letters*, 12(11):2183–2187. 49

Yu, Y., Guan, H., Zai, D., and Ji, Z. (2016). Rotation-and-scale-invariant airplane detection in high-resolution satellite images based on deep-hough-forests. *ISPRS Journal of Photogrammetry and Remote Sensing*, 112:50–64. 49

Yuan, Q., Shen, H., Li, T., Li, Z., Li, S., Jiang, Y., Xu, H., Tan, W., Yang, Q., Wang, J., Gao, J., and Zhang, L. (2020). Deep learning in environmental remote sensing: Achievements and challenges. *Remote Sensing of Environment*, 241(February):111716. 14

Yuan, X., Shi, J., and Gu, L. (2021). A review of deep learning methods for semantic segmentation of remote sensing imagery. *Expert Systems with Applications*, 169(November 2020):114417. 10

Zamir, S. W., Arora, A., Gupta, A., Khan, S., Sun, G., Khan, F. S., Zhu, F., Shao, L., Xia, G. S., and Bai, X. (2019). isaid: A large-scale dataset for instance segmentation in aerial images. pages 28–37, Long Beach, CA, USA, USA. 69

Zeng, Y., Duan, Q., Chen, X., Peng, D., Mao, Y., and Yang, K. (2021). Uavdata: A dataset for unmanned aerial vehicle detection. *Soft Computing*, 25(7):5385–5393. 69, 91

Zhang, D., Song, Y., Liu, D., Jia, H., Liu, S., Xia, Y., Huang, H., and Cai, W. (2018a). Panoptic segmentation with an end-to-end cell r-cnn for pathology image analysis. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 237–244. Springer. 106

Zhang, L., Zhang, L., and Du, B. (2016). Deep learning for remote sensing data: A technical tutorial on the state of the art. *IEEE Geoscience and Remote Sensing Magazine*, 4(2):22–40. 14

Zhang, S., He, G., Chen, H.-B., Jing, N., and Wang, Q. (2019). Scale adaptive proposal network for object detection in remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, 16(6):864–868. 25

Zhang, X., Zeraatpisheh, M., Rahman, M. M., Wang, S., and Xu, M. (2021). Texture is important in improving the accuracy of mapping photovoltaic power plants: A case study of ningxia autonomous region, china. *Remote Sensing*, 13(19). 30

Zhang, X. and Zhu, X. (2019). An efficient and scene-adaptive algorithm for vehicle detection in aerial images using an improved yolov3 framework. *ISPRS International Journal of Geo-Information*, 8(11):483. 52

Zhang, Y., Qiu, Z., Yao, T., Liu, D., and Mei, T. (2018b). Fully convolutional adaptation networks for semantic segmentation. pages 6810–6818, Salt Lake City, UT, USA. IEEE. 82

Zhang, Z., Geiger, J., Pohjalainen, J., Mousa, A. E.-D., Jin, W., and Schuller, B. (2018c). Deep learning for environmentally robust speech recognition. *ACM Transactions on Intelligent Systems and Technology*, 9(5):1–28. 1, 14

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890. 57

Zhao, W., Persello, C., and Stein, A. (2021). Building outline delineation: From aerial images to polygons with an improved end-to-end learning framework. *ISPRS Journal of Photogrammetry and Remote Sensing*, 175(September 2020):119–131. 23

Zhao, Z.-Q. Q., Zheng, P., Xu, S.-T. T., and Wu, X. (2019). Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, 30(11):3212–3232. 1, 14, 47

Zheng, Z., Zhou, G., Wang, Y., Liu, Y., Li, X., Wang, X., and Jiang, L. (2013). A novel vehicle detection method with high resolution highway aerial image. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 6(6):2338–2343. 48, 49

Zhong, J., Lei, T., and Yao, G. (2017). Robust vehicle detection in aerial images based on cascaded convolutional neural networks. *Sensors (Switzerland)*, 17(12). 52

Zhou, H., Wei, L., Lim, C. P., Creighton, D., and Nahavandi, S. (2018a). Robust vehicle detection in aerial images using bag-of-words and orientation aware scanning. *IEEE Transactions on Geoscience and Remote Sensing*, 56(12):7074–7085. 50

Zhou, T., Ruan, S., and Canu, S. (2019). A review: Deep learning for medical image segmentation using multi-modality fusion. *Array*, 3-4:100004. 1, 14

Zhou, Z., Rahman Siddiquee, M. M., Tajbakhsh, N., and Liang, J. (2018b). Unet++: A nested u-net architecture for medical image segmentation. In Stoyanov, D., Taylor, Z., Carneiro, G., Syeda-Mahmood, T., Martel, A., Maier-Hein, L., Tavares, J. M. R., Bradley, A., Papa, J. P., Belagiannis, V., Nascimento, J. C., Lu, Z., Conjeti, S., Moradi, M., Greenspan, H., and Madabhushi, A., editors, *Miccai*, volume 11045 of *Lecture Notes in Computer Science*, pages 3–11. Springer International Publishing, Cham. 35, 112

Zhu, J., Sun, K., Jia, S., Li, Q., Hou, X., Lin, W., Liu, B., and Qiu, G. (2018). Urban traffic density estimation based on ultrahigh-resolution uav video and deep neural network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(12):4968–4981. 52

Zhu, X. X., Tuia, D., Mou, L., Xia, G.-s., Zhang, L., Xu, F., and Fraundorfer, F. (2017). Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine*, 5(4):8–36. 14

Zhuang, L., Zhang, Z., and Wang, L. (2020). The automatic segmentation of residential solar panels based on satellite images: A cross learning driven u-net method. *Applied Soft Computing Journal*, 92:106283. 30