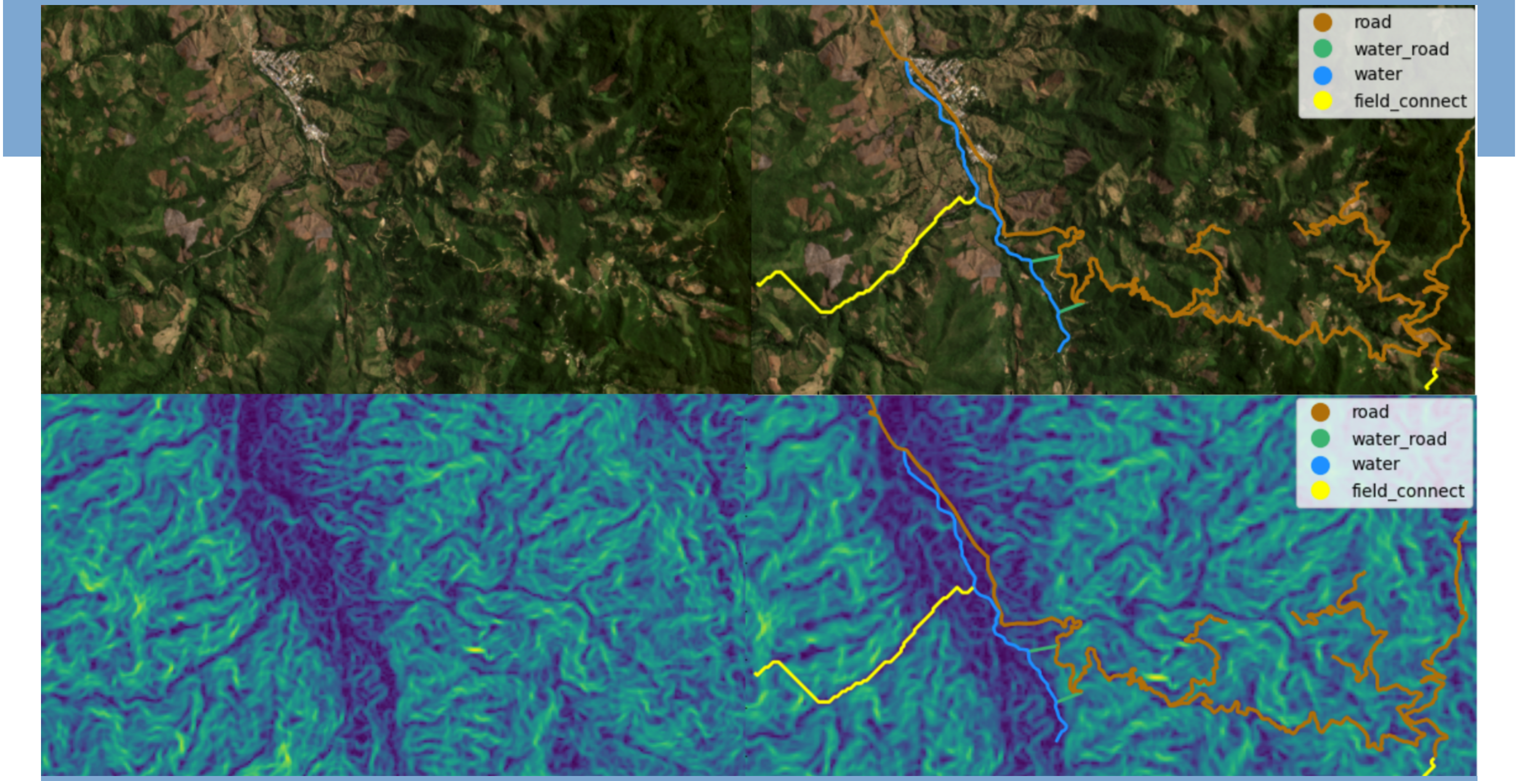


Geospatial Intelligence for Identifying Supply Chains with Satellite Data and Machine Learning

H. Denekamp
MSc Thesis
2024-12-10

Delft University of Technology
Faculty of Technology, Policy & Management
Engineering and Policy Analysis

Geospatial Intelligence for Identifying Supply Chains with Satellite Data and Machine Learning

Hanna-Lieske Denekamp
4441761

to obtain the degree of Master of Science
at the Delft University of Technology

Thesis Committee:
Prof.dr.ir. A. Verbraeck - Delft University of Technology
Dr. I. Lefter - Delft University of Technology
Ir. I.M. van Schilt - Delft University of Technology

Faculty of Technology, Policy & Management
Engineering and Policy Analysis

Data and model available through:
`doi.org/:10.4121/84da525c-ecc3-4fa3-8742-011b4c503f20`
`https://github.com/hannadenekamp/GEOAI_supply_chains`

Example of a node connected to graph through pathfinding
Image adapted from Planet Labs PBC (2024b)

Summary

This research explores the application of geospatial intelligence and machine learning to map and analyse routes used in illicit supply chains, specifically targeting the networks that transport illegal goods from South America to Europe. With drug trafficking routes becoming more adaptive and complex, particularly in the corridors connecting South America to Europe, traditional methods of monitoring and interception have proven insufficient. Enforcement strategies often rely on seizure data, which is delayed and offers an incomplete view of these evolving networks. To address this, the study proposes a novel methodology that combines satellite data with machine learning to identify and map elements of the supply chain, such as cultivation fields, clandestine laboratories, and transportation routes. This approach aims to create a scalable solution for tracking these illicit networks and supporting law enforcement.

Through the use of supervised machine learning models, potential cultivation fields and clandestine laboratories are detected in satellite imagery. The methodology includes sourcing data, configuring machine learning models based on literature, and evaluating model performance. The challenge in training models is the limited labelled data for illegal activities.

Once these nodes are identified, a connected graph is constructed that integrates both licit and illicit transport routes. This graph serves as the foundation for estimating smuggling routes. The process involves combining graphs of waterways and roads with airstrips, ports, potential cultivation fields and laboratories. To estimate paths between illicit sites and the main infrastructure network, an algorithm is applied that estimates the least steep path based on a slope raster.

Route estimation is the next step, where Dijkstra's algorithm is employed to calculate the most probable paths. Each edge is weighted based on travel time, cost, and risk, with factors adjusted to account for various modes of transport, including trucks, boats, planes, and donkeys, depending on terrain. There is also a factor added to account for transloading. These weighted paths offer a realistic picture of choices traffickers might make to minimize travel time, cost or risk.

To test the model, a case study is performed focusing on the Colombia-Venezuela border region, a region with high trafficking activity. The case study applies the graph-based methodology to identify plausible trafficking routes, connecting cultivation sites to ports. By calculating optimal paths, the study reveals which routes are likely to be used most frequently, providing insights that could potentially aid in targeted enforcement.

There are some limitations and areas for future improvement in this proof of concept, such as challenges with data quality and model accuracy, as well as potential refinements to the methodology to enhance real-world applicability. However, there is potential in combining machine learning with geospatial intelligence to create a tool for tracking and disrupting illicit supply chains. This research offers a proof of concept for enabling scalable monitoring of illegal activities, supporting efforts to disrupt global trafficking networks.

Contents

List of Figures

List of Tables

1 | Introduction

Since 2015, Europe has experienced a significant rise in the use of illicit substances. This rise can be attributed to increased availability and decreased prices (UNODC, 2022). This surge in the illicit substance market leads to heightened drug-related violence, medical complications, and mortality, thereby destabilizing European society. The growing supply chain between South America and Europe has been identified as a primary factor contributing to the increased availability (UNODC, 2022).

Figure 1.1 shows a simplification of the supply chain for illicit substances from South America to Western countries. Sourcing, manufacturing, distribution, retail, and consumption are the general segments of a supply chain (Vermeulen et al., 2018). This supply chain begins in South America, where the sourcing and manufacturing occur. The processed substances are subsequently distributed to other regions, including Europe (UNODC, 2022). In Europe, the substances are further distributed, retailed and consumed.

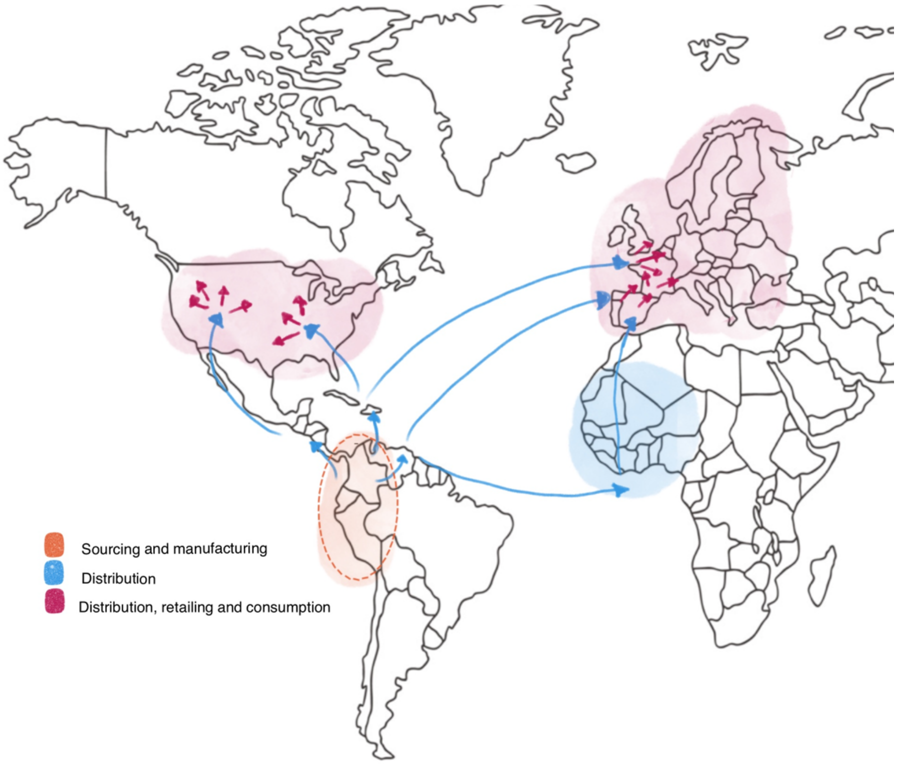

Sourcing and manufacturing
Distribution
Distribution, retailing and consumption

Figure 1.1: Generalized supply chain of illicit substances from South America to Western countries

Comprehensive knowledge of these supply chains remains limited, while this is vital for disrupting the illicit supply chain and limiting its destabilizing effects on society. The research that has been done, e.g. Magliocca et al. (2019), identified the illicit supply chain as a complex adaptive system. It consists of multiple smuggling networks, each emergent, self-organized and highly adaptive systems. Their evolution and spatial relocation result from past interdiction events. Increased enforcement in one area often leads to shifts in distribution routes, transportation methods, and sourcing and manufacturing locations. As many groups and individuals operate in the supply chain, when one entity is apprehended, another quickly takes its place. This dynamic nature makes it difficult for governmental organizations to effectively trace and disrupt these supply chains.

A deeper understanding of these illicit networks and their dynamics is required to design effective disruption strategies. Unlike legal supply chains, data on the logistics of the illicit supply chain is sparse (e.g. Vermeulen et al., 2018; van Schilt et al., 2024a,b). Estimations of the flow of illicit goods are primarily based on seizure data, as done by the United Nations Office on Drugs and Crime (UNODC, 2022). However, this data is collected annually, and there is a delay before publication. Due to the adaptive supply chain, the data may be outdated upon publication. Additionally, seizure data is biased, as it only reflects areas where enforcement checks are conducted.

There is a need for methods to obtain insights into the locations and routes where goods are produced, manufactured, and distributed. Although these activities may be illicit, they cannot be completely concealed, as observations and local intelligence can potentially reveal them. Satellite data is a relatively new tool for obtaining intelligence in large areas (Pinto Hidalgo & Silva Centeno, 2022), which can be particularly useful for the initial stages of the supply chain: sourcing and manufacturing. The field of combining satellite data with other sources of data to gain an understanding of geospatial activities is called geospatial intelligence.

1.1 Defining geospatial intelligence

Before defining geospatial intelligence, it is important to define intelligence itself. Intelligence is the evaluation, analysis, and presentation of data, information, and knowledge in a decision-making format (OSCE, 2017). Data, information, and knowledge are defined as the following:

Data are raw observations and measurements. For example, the satellite imagery.

Information is this data put into context. This is, for example, recognizing a spatial pattern in satellite imagery.

Knowledge is information that has been interpreted and understood. For example, when an expert adds their insights to the recognised pattern.

Geospatial intelligence (GEOINT) refers to analysing satellite imagery and other georeferenced observations and measurements as data to describe, assess and visualize geo-referenced features and activities on earth (NGA, 2018). Integrating data from multiple sources is intrinsic to GEOINT. To integrate the data, it uses and combines technologies, such as remote sensing, artificial intelligence and human expertise (Pinto Hidalgo & Silva Centeno, 2022). Geospatial Artificial Intelligence (GEOAI) is a sub-discipline of GEOINT, where machine learning (ML) is used to extract and analyse information from large geospatial datasets (UN GGIM, 2015).

GEOINT and GEOAI are particularly suitable for this research for several reasons. Firstly, the activities within the supply chain are inherently spatial (Magliocca et al., 2019), requiring geospatial analysis to track and understand them effectively. Furthermore, satellite imagery has the potential to be a comprehensive resource for monitoring these activities (Escobar-López et al., 2022; Pinto Hidalgo & Silva Centeno, 2022; Anderson & Potter, 2022). Machine learning techniques enhance this process by identifying objects in satellite imagery related to illicit activities within the supply chain, transforming raw data into information. Additionally, the physical infrastructure, such as roads, potentially used for smuggling within the supply chain can be mapped and analysed using geospatial data sources. Integrating expert knowledge allows for interpreting this information and converting it into deeper insights into the illicit supply chain dynamics.

1.2 Identification of relevant literature

A systematic search was conducted using the keywords listed in Table 1.1. The primary focus was on papers addressing topics related to illicit activities or objects. This targeted approach yielded a relatively small number of relevant studies, reflecting the narrow scope of the search. However, this focus was intentional. For routes, the lack of accessible data in illicit supply chains necessitated an alternative approach distinct from those applied to licit supply chains. For machine learning, the emphasis on illicit topics was driven by the objective of identifying datasets and models specifically applicable to the detection of illicit objects. This strategy ensured alignment with the study's objectives, despite the limited availability of related research.

Preposition	**Topic**	**Data**	**Techniques**	**Location**
illicit	supply chain	GIS	machine learning	South America
illegal	production	remote sensing	CNN	Colombia
criminal	trade	satellite data	classification	Latin America
	networks	sentinel data	GEOAI	Venezuela
	land use cover	satellite imagery	GEOINT	
	cultivation	nighttime light data		
	crops			
	airstrips			
	flights			
	laboratories			
	distribution centres			
	ports			
	infrastructure			
	transport			
	route choice			
	trafficking			
	smuggling			
	pathfinding			

Table 1.1: Keywords used in literature search

1.3 Illicit supply chains

This section describes the illicit supply chains in three subsections. The first subsection 1.3.1 describes the identification of illicit supply chains. The second subsection 1.3.2 describes the graph representation of illicit supply chains. The third subsection 1.3.3 gives an overview of illicit components in the supply chain and how to detect them with satellite data and GEOAI.

1.3.1 Route identification

In a supply chain, goods are transported via specific routes. Traditionally, route choice prediction relies on models that range from simple statistical decision-making models to sophisticated computerized models for complex scenarios. These models depend on data derived from documentation, sensors, or surveys. However, such data are typically unavailable or severely limited for smuggling routes, leaving the field of route identification in illicit supply chains under-researched. In the limited studies conducted, data have been gathered from diverse sources (see Table A.1 in Appendix A), with spatial characteristics such as terrain, police presence, and population density playing a critical role in shaping smuggling routes (Magliocca et al., 2022).

Two main approaches for identifying routes in illegal supply chains can be distinguished. The first approach focuses on estimating the attractiveness of specific areas for smuggling using techniques such as multi-criteria decision-making (Robert et al., 2015), mixed-effects models (Magliocca et al., 2022), and tree-based classification approaches (Niamkaeo & Robert, 2020). The second approach leverages graph theory to estimate routes. For example, Achi et al. (2012) employed a graph of official road networks, integrating spatial characteristics to identify potential illicit destinations. Using this framework, a shortest-path algorithm was applied to determine probable routes. Similarly, Magliocca et al. (2019) utilized an agent-based model on an aggregated international network, where agents selected routes based on interdiction risks influenced by spatial features.

This study focuses on observable infrastructure, such as those identified through satellite imagery, making the use of non-aggregated graphs particularly relevant. Given the computational intensity of agent-based models when applied to extensive, non-aggregated networks, shortest-path algorithms appear as a more practical alternative. Nevertheless, path selection should account not only for distance but also for factors like interdiction risks and associated costs (Klaassen, 2021). This necessitates assigning appropriate edge weights within the graph to reflect these considerations.

1.3.2 Graph representation

To model the illicit substance supply chain as a graph, the infrastructure must be categorized into nodes and edges. In this framework, routes represent the pathways through which substances are transported using various modalities. This study focuses specifically on smuggling routes leading to ports in Europe.

- An illicit supply chain can be conceptualized as comprising two main components:
  - A graph (infrastructure)
    - Nodes (licit and illicit)
    - Edges (licit and illicit)
  - A path through the graph (smuggling routes)

Aschner & Montero (2021) distinguished two categories of infrastructure within illicit supply chains: production and distribution infrastructure and support infrastructure. The production and distribution infrastructure is the primary focus of this study, as it directly involves the transportation of illicit substances. This infrastructure contains licit structures and purpose-built illicit components (Aschner & Montero, 2021). The relevant part of the supply chain is further categorized into the nodes and edges, shown in table 1.2

Table 1.2: Nodes and Edges in the Illicit Supply Chain

Nodes	**Edges**
Cultivation fields	Roads (licit and illicit)
(Clandestine) airstrips	Waterways
(In)formal distribution centres	Airways (licit and illicit)
(Clandestine) inland ports	
Seaports	

1.3.3 Detection with satellite data and Geospatial Artificial Intelligence

GEOAI can aid in identifying observable components of the supply chain, specifically the nodes and edges. While data on licit infrastructure, such as official roads and waterways, is publicly available, data on illicit infrastructure remains scarce.

A literature search on classifying and identifying illicit infrastructure using satellite imagery and machine learning has been conducted, of which an overview is given in Table A.2 in Appendix A. Given the limited research on illicit infrastructure, some studies on licit infrastructure are also considered relevant.

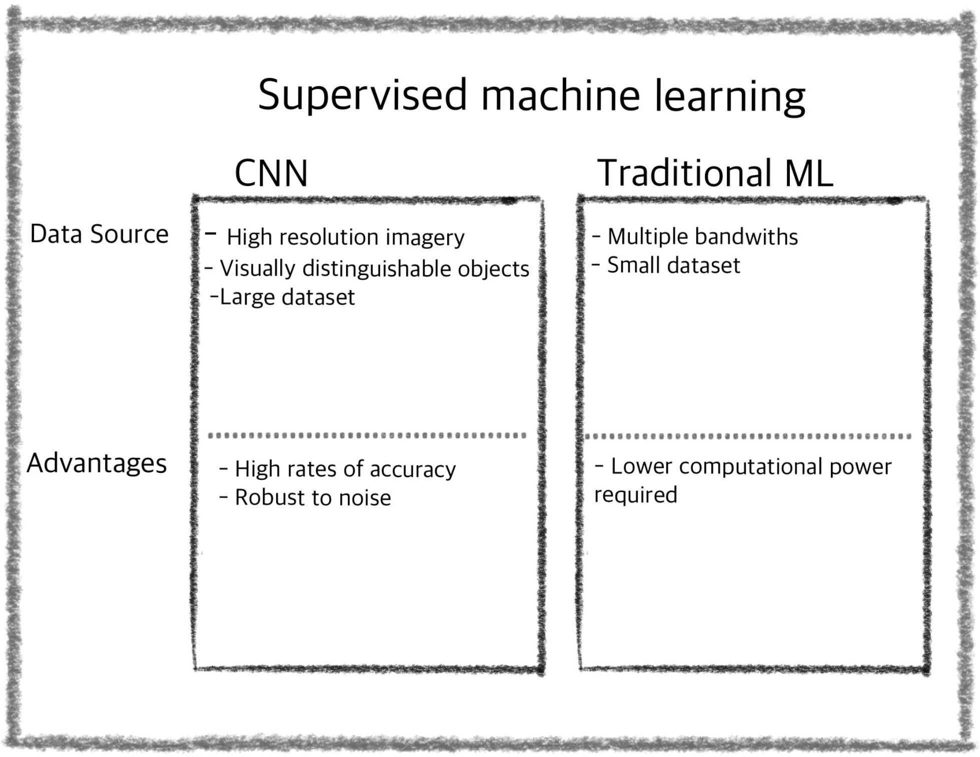


Figure 1.2: Comparison of CNN and traditional Machine Learning

A comparison between convolutional neural networks (CNNs) and traditional ML-techniques, based on the literature presented in Table A.2, is provided in Figure 1.2. Most studies employ supervised machine learning models, which require labelled training datasets. The model selection depends on the analysis's specific objectives and the characteristics of the satellite data or training dataset. Convolutional neural networks are commonly used to identify visually distinctive objects, such as features in high-resolution satellite imagery. However, CNNs are computationally intensive, which can be a limiting factor. In cases where computational speed is a priority, more traditional machine learning approaches are often preferred, particularly when multiple wavelengths are used to identify objects that are difficult to distinguish visually. The subsequent paragraphs delve into the application of satellite data and machine learning techniques for the detection of illicit components within the supply chain.

Cultivation fields

Three methods can be distinguished in the literature for detecting cultivation. The first method labels each so-called chip, a raster with a pixel x pixel size, of satellite imagery with land use (Shendryk et al., 2019). The second method involves land use cover classification, where sections of satellite imagery are classified as specific land uses, such as cultivation (Nazarova et al., 2020; Bolívar-Santamaría & Reu, 2021; González-González et al., 2022). The third method is identifying a specific type of cultivation, where only sections with that particular type of cultivation are labelled (Escobar-López et al., 2022; Berkson et al., 2020).

The first two methods do not distinguish between licit and illicit cultivation. However, they can still be useful for identifying illicit fields if combined with other land characteristics to assess the likelihood of licit or illicit cultivation. The third method is the most accurate for detecting a specific field type, but it requires a labelled training set of illicit cultivation fields.

Laboratories

Laboratories for producing illicit substances can be categorized into two types: small kitchens, typically located in small buildings or sheds away from habitation, and larger laboratories situated closer to inhabited areas. Models designed to detect general habitation or buildings are inadequate for identifying laboratories. For the small kitchens, this is due to their distinct features and locations, while for the larger laboratories, this is due to visual similarity to other buildings. Notably, only one study has addressed the identification of illicit laboratories using satellite data. Pinto Hidalgo & Silva Centeno (2022) developed a dataset using PlanetScope imagery, and they trained a Convolutional Neural Network (CNN) to identify buildings that are potential laboratories.

Illicit airstrips

For the purpose of detecting illicit airstrips, models that focus on detecting official airstrips are not suitable. Illicit airstrips are often unpaved and located in a forested area with little habitation, whereas licit airstrips are typically paved and surrounded by buildings. Only one study has addressed the detection of illicit airstrips (Becerra et al., 2021), but their data and model are not publicly available. Additionally, some journalistic research has been conducted on illicit airstrips (Anderson & Potter, 2022). This research involved manually examining images and verifying locations with helicopters to determine where illicit airstrips are situated. The journalists utilized this data as input for machine learning models to detect illegal mines. However, this dataset could be used to train a model to identify illicit airstrips in general.

Distribution centres and inland ports

Research on the identification of distribution centres and inland ports is limited. Identifying distribution centres can be hard, as they are often not visually distinguishable from other distribution buildings, or they have sometimes even been built underground. Identifying small, possibly, illicit ports with satellite data has more potential. However, the sole study on detecting illicit ports is by Becerra et al. (2021), whose data and model are not publicly available. Given that ports and distribution centres are at the end of the supply chain within the scope of this research, pinpointing their exact locations may not be necessary; assigning approximate locations based on other data could be sufficient.

Unofficial roads

A differentiation can be made between licit and illicit roads. Licit roads are roads documented on official maps, while illicit roads are illegally created to enable illicit activities. Licit roads do not require detection, as they are well documented. In contrast, illicit roads, which are not documented, can be detected using satellite data and machine learning. Botelho et al. (2022) developed a road detection model for the Brazilian Amazon to map these unofficial roads using Sentinel-2 data. Since unofficial roads are often unpaved, they are more challenging to identify. However, the use Sentinel-2 data with a U-net algorithm, a type of CNN, has proven suitable for this purpose (Botelho et al., 2022).

1.4 Research questions

This research aims to develop a proof of concept for a method that utilizes open-source data to estimate smuggling locations and routes. This research's main focus will be the sourcing, manufacturing, and distribution network in South America, where we will study plausible routes from cultivation to an international port.

This leads to the primary research question:

How can satellite data combined with machine learning models be used to identify plausible routes based on illicit and licit infrastructure network?

The following sub-questions are formulated:

- Q1. To what extent are machine learning model(s) suitable for identifying plausible illicit nodes of the infrastructure graph given the open-source satellite imagery and georeferenced data?
- Q2. How can these illicit nodes and vertices be combined with licit infrastructure into a graph?
- Q3. How can plausible routes be identified given the identified combined graph of the licit and illicit infrastructure?
- Q4. Given this approach, what are the plausible routes based on the identified illicit and licit infrastructure network in the border region of Colombia and Venezuela?

Figure 1.3 shows the proposed method to answer the research questions. To address the research questions, algorithms will be developed in Python, due to its libraries that support data preparation for imagery data, machine learning, visualization, and routing algorithms. Chapter 2 will further elaborate on the concept of the methods and demarcation. The second chapter will answer the first research question on detecting illicit objects with machine learning. Chapter 4 answers the second research question on combining licit and illicit nodes and edges into one graph. The third research question is answered in Chapter 5, where the method for estimating routes based on edge weights will be explained. A case study is then performed to test the proposed model, which is documented in Chapter 6.

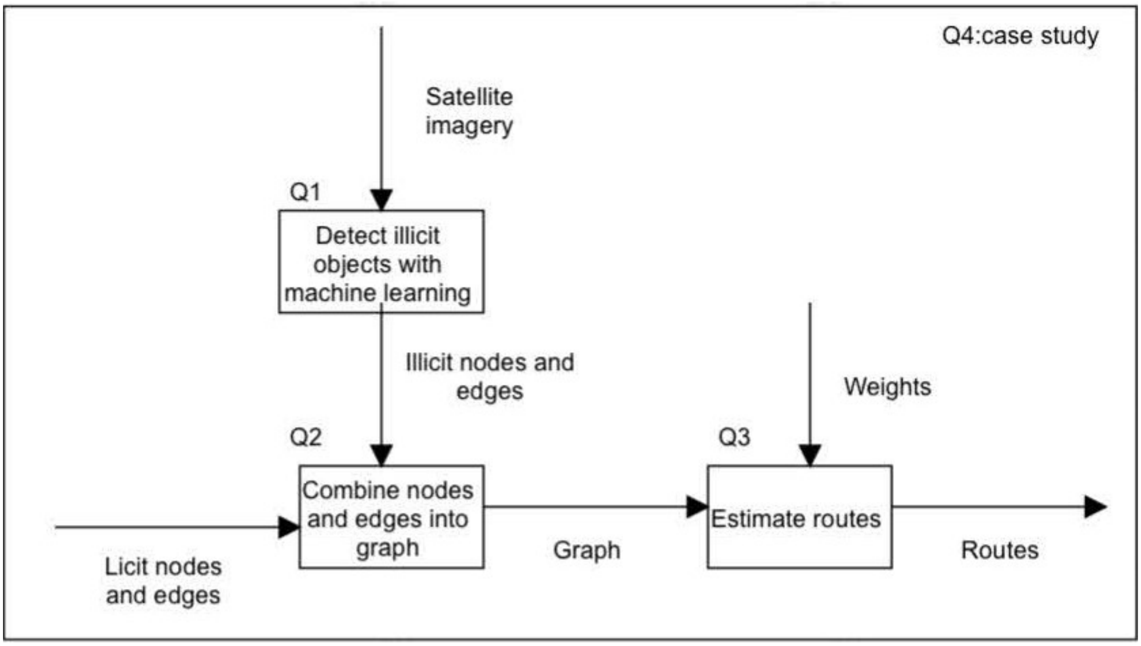


Figure 1.3: Research approach

2 | Illicit supply chain conceptualisation

This chapter describes the conceptualisation for the illicit supply chains. The first section 2.1 gives a brief overview of the illicit supply chain and it's components in South America. The second section 2.2 describes the methodology for supply chain identification.

2.1 Components of the illicit supply chain in South America

Pinto Hidalgo & Silva Centeno (2022) describe the process of illicit cultivation and production in South America. The production of illicit goods begins in agricultural fields where the crops are cultivated. These raw materials are transported to clandestine laboratories. In these laboratories, the raw leaves undergo initial processing to produce a paste, which is then refined into a base product. This process may take place in a single, larger laboratory or be distributed across smaller laboratories, kitchens, and larger facilities. The base product is later transported to distribution points, from where it is eventually transported to ports for further distribution. Different transportation modalities are used, including donkeys, trucks, boats, and planes. The selection of a specific transportation method depends on factors such as terrain, risk, cost, and travel time (Figure 2.1).

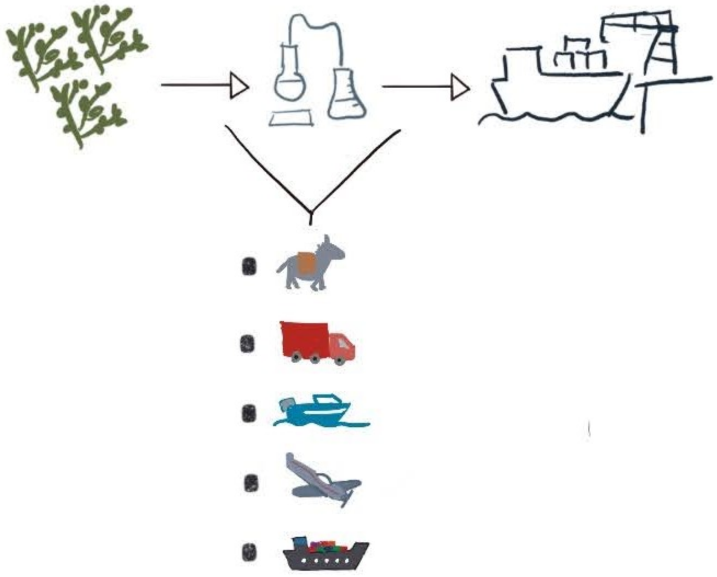


Figure 2.1: Illicit supply chain in South America with modalities

Identifying every illicit component of the supply chain is not feasible due to time constraints and the limitation of available models and labelled data. Figure 2.1 illustrates a simplified version of the supply chain in South America. The focus is on the most critical elements that are identifiable and essential to understanding the supply chain. This leads to the following aggregation of the supply chain in nodes and edges:

Nodes

Potential cultivation fields: Areas identified as likely locations for the cultivation of illicit crops

Potential laboratories: Objects identified as likely small, clandestine laboratories located in uninhabited or remote areas

Potentially clandestine airstrips: Airstrips that may be used illicitly from OpenStreetMap (OSM) data

Seaports: Small and large ports

Edges

Roads: Official roadways that may be used for transporting goods. Some illicit roads may be included if they are recorded in OSM

Waterways: Rivers and canals used as transportation routes

Airways: Air routes connecting potentially clandestine airstrips

The algorithms have to focus on potential nodes and edges rather than nodes and edges that are completely certain, due to the inherent uncertainties in the available models, which cannot conclusively identify the objects. By narrowing the scope to include only licit roads and excluding distribution centres and inland ports due to the lack of available datasets and models, the research remains manageable while still providing proof of concept.

Although there is a dataset available for clandestine airstrips (Anderson & Potter, 2022), this dataset requires a lot of manual data preparation, and there is no trained machine learning model available. Alternatively, potentially clandestine airstrips can be sourced from OSM, providing a quicker, albeit less comprehensive, option sufficient for the proof of concept.

Illegal distribution centres are not included due to the difficulty of visually detecting them with machine learning models and the lack of another existing method to include them. Illegal inland ports are not included as well, due to the lack of availability of a trained model or dataset.

2.2 Methodology for identification

A proof of concept for a method to estimate smuggling locations and routes must be developed. The conceptualization of this method is depicted in Figure 2.2. The construction of a representative supply chain model involves several stages, each focused on integrating diverse data sources.

The **first phase** is identifying potential cultivation fields and laboratories using machine learning models. Potential cultivation fields are detected using a model trained on the dataset provided by Goldenberg et al. (2017). This dataset contains labelled land-use chips from the Amazon region. These outcomes are further filtered using elevation data to isolate regions most likely to be involved in illicit cultivation. Laboratories are identified using a similar approach. The coca-paste detection dataset, which provides bounding box labels for potential laboratory locations, is suitable for this task (Pinto, 2022).

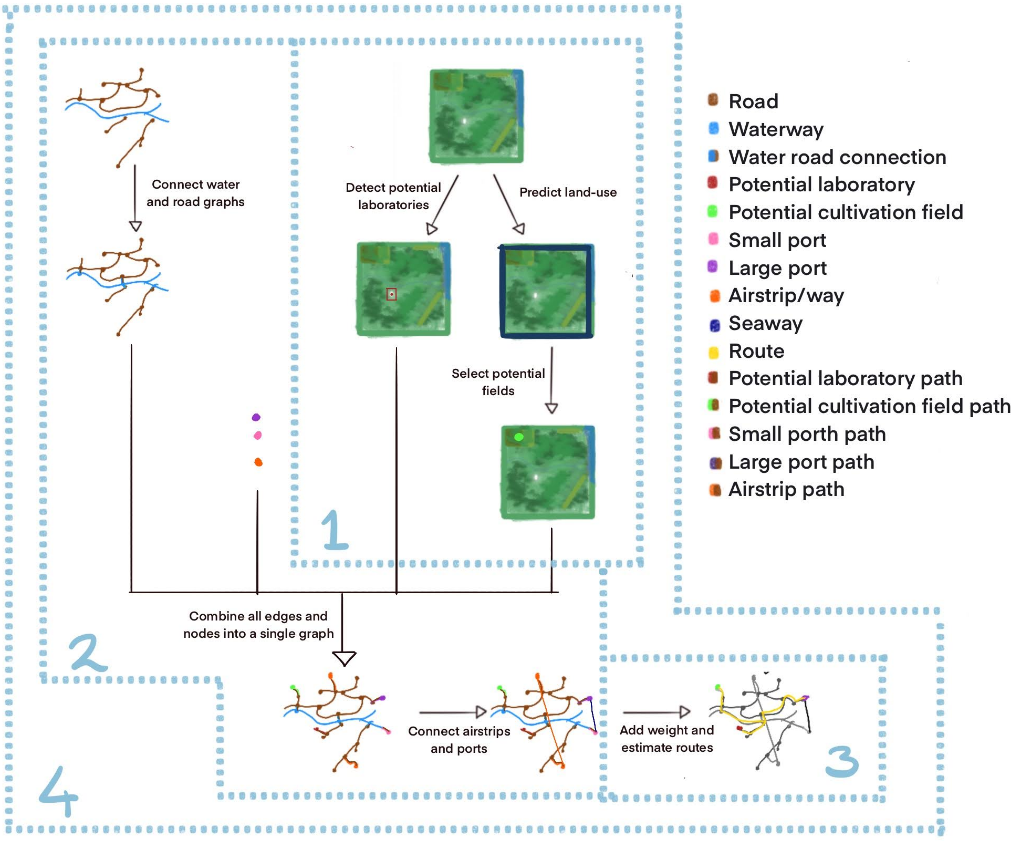

Connect water and road graphs
Detect potential laboratories
Predict land-use
Select potential fields
1
Combine all edges and nodes into a single graph
2
Connect airstrips and ports
Add weight and estimate routes
3
4
Road
Waterway
Water road connection
Potential laboratory
Potential cultivation field
Small port
Large port
Airstrip/way
Seaway
Route
Potential laboratory path
Potential cultivation field path
Small porth path
Large port path
Airstrip path

Figure 2.2: Conceptual method for illicit supply chain and route identification

The **second phase** involves constructing a graph with licit and illicit components. It starts with connecting the separate water and road graphs. These graphs can be obtained from OSM using the OSMNX library in Python. For the proof of concept, it is sufficient to include only the licit roads, but some illicit roads might automatically be included due to user-generated contributions to OSM.

The identified potential laboratories and fields, airstrips and ports are added to the graph. Ports, essential for the final stages of the supply chain, will be incorporated from an appropriate dataset.

Illicit nodes are often not directly connected to the official road network but are linked via illicit roads. Since these illicit roads will not be directly identified through machine learning, their paths can be estimated using a slope raster to determine the least steep path between the illicit nodes and the nearest road.

The last step in constructing the graph is establishing connections between the network and airstrips and ports, ensuring that each subgraph is internally connected. The airstrips will be included in a fully connected (sub)graph, since airplanes can fly between any two locations. The ports will be connected based on actual shipping data.

The **third phase** involves attributing weights to the graph with various modalities (donkeys, trucks, boats, planes) and their corresponding costs, travel times, and risks. Using a shortest path algorithm, optimal routes from fields to laboratories to ports will be calculated for each criterion (cost, travel time, and risk).

The **fourth phase** involves applying this proposed model on a case study in South America. Specifically, it will zoom in on the border region between Colombia and Venezuela. This region is linked to the illicit supply chain to Europe (UNODC, 2022; Pacheco-Pascagaza et al., 2022) and possesses characteristics that attract traffickers, such as sparse population, forestry, and available water sources.

3 | Identifying potential illicit nodes

Supervised machine learning has proven to be an effective tool in identifying objects and areas associated with illicit activity in satellite imagery (Berkson et al., 2020; Pinto Hidalgo & Silva Centeno, 2022; Becerra et al., 2021; Botelho et al., 2022; Emily & Sudha, 2022). This research will focus on identifying potential cultivation fields and laboratories, as mentioned in the conceptualisation chapter. Although traditional machine learning methods are suitable, CNNs are often the preferred method for computer vision tasks, such as object detection and image segmentation (Emily & Sudha, 2022). The CNN is a trainable architecture consisting of typically one to three feature extraction stages. Each stage has three layers: a convolutional layer, a nonlinearity layer, and a pooling layer. After the feature-extraction stages, there are one or more traditional, fully connected layers and a final classifier layer (Zhang et al., 2016). This architectural design enables CNNs to extract high-level abstract features from the original pixel values of images. This chapter describes how these convolutional neural networks are used to detect potential fields in the first section 3.1 and laboratories in the second section 3.2.

3.1 Detection of potential fields

3.1.1 Dataset

The initial step in identifying potential cultivation fields involves selecting an appropriate dataset. Berkson et al. (2020) have created a labelled dataset with coca fields, but this dataset is not publicly accessible and, therefore, cannot be used in this research. The next best option is a dataset provided by Goldenberg et al. (2017). This dataset contains 40000 training chips and 40000 testing chips of the Amazon with a spatial resolution of 3 to 5 meters and an approximate area of 1 km^2 per chip. The chips are RBG .png files. Figure 3.1 shows an example of the labels.

The dataset is particularly suitable for this research as it includes cultivation labels and covers the Amazon region, which is part of the region where illicit cultivation occurs. There are, however, some limitations of this dataset. The dataset does not include the Andes region, which has more elevation differences than the Amazon region. It also does not offer specificity in the type of cultivation or the specific cultivation area, but with further filtering with elevation data, it is possible to isolate regions most likely to be involved in illicit cultivation.

The dataset contains a vast class imbalance, with primary forest presence on more than 90% of the chips, whereas conventional mining only accounts for 0.25% (Figure 3.2. Cultivation is, for this research, the most important label, which is present on 11.37% of the chips. It is important to monitor the model's performance for these rarer classes, as it is possible to get a good overall performance while the model does not predict well for the rare classes.

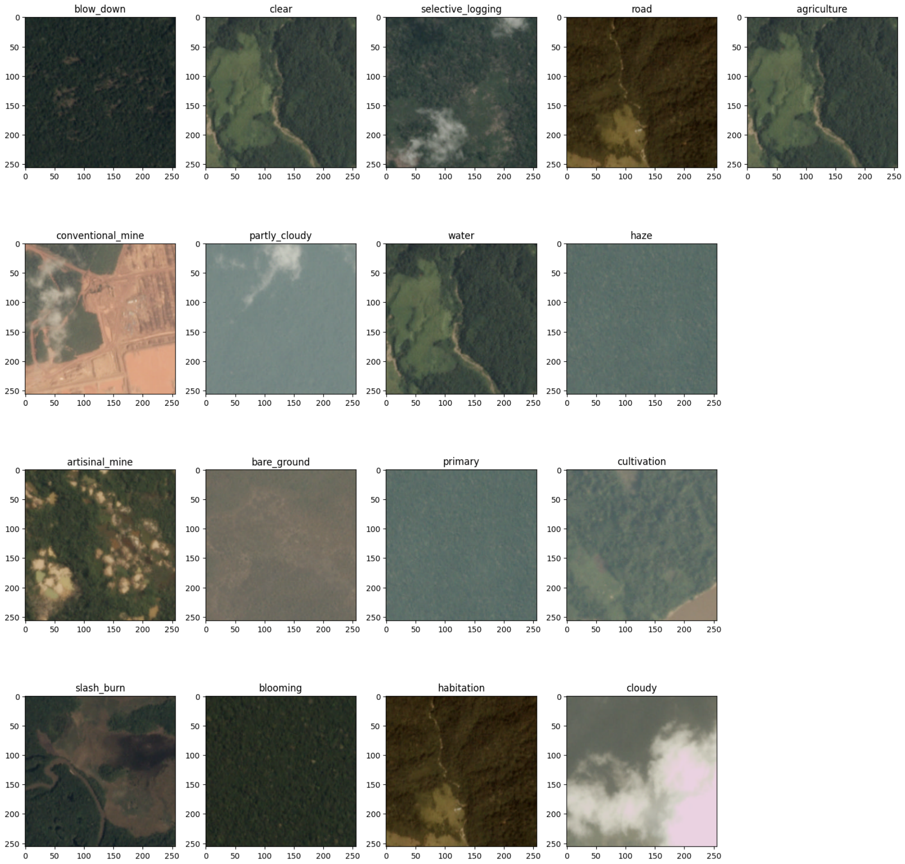


Figure 3.1: Examples of chips for each label

3.1.2 Classification procedure

Due to the large size of the dataset, a CNN is very suitable. There are many different CNN architectures. A CNN model in Python by Maladière (2022) with a ResNet-18 architecture is used because of its good performance, clear documentation and the ability to train the model on a single computer.

The code and design choices from Maladière (2022) were used in this project. We employed the ResNet-18 model, which was implemented in Python using PyTorch (Ansel et al., 2024). For optimal performance, ResNet-18 requires input shapes that are multiples of 32. The image input size is 256, and the closest multiple of 32 is 8. For no particular reason, the input data is resized to 224, which is also a multiple of 32, and normalized based on the mean and standard deviation of the pre-trained ResNet model. Data augmentation is used during the training and testing phase. Test-time augmentation (TTA) enhances the robustness of the model by applying random transformations to each image at execution time without increasing the size of the dataset. The dataset is split into a training set (80% of the images) and a validation set (20% of the images).

First, the target encoder is defined, and then the custom training and validation datasets are wrapped within DataLoaders with a batch size of 64, which balances RAM usage and speed. Pre-trained ResNet-18 weights are downloaded and all layers are frozen except the last fully connected layer. This final layer has two dense layers followed by a sigmoid activation function. This newly added fully connected (fc) layer is the only part of the trained model.

The model is trained for 21 epochs, decreasing the learning rate by a factor of 10 every 7 batches. The primary metric is the validation loss. The validation Fbeta-score is only used as a secondary indication, since the classification threshold is arbitrarily

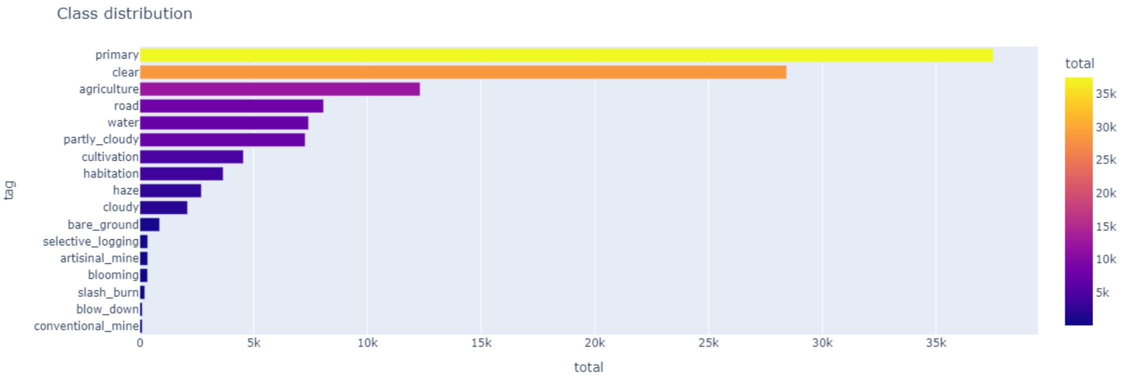


Figure 3.2: Distribution of labels in dataset

set at 0.2. Later, the optimal threshold for each target class is determined and the final Fbeta-score is determined.

3.1.3 Performance of the classification model

As a performance measure of the model, Binary Cross-Entropy (BCE) Loss is chosen, which is suitable for models that output probabilities for binary classification tasks (Alake, 2023). The BCE Loss measures how far off a model's predicted probability is from the actual labels. The BCE loss function penalizes inaccurate predictions, which are predictions that have a significant difference from the positive class. This encourages the model to refine its predictions. The mathematical loss function is as follows:

$$L(y, f(x)) = -(y * \log(f(x)) + (1 - y) * \log(1 - f(x)) \tag{3.1}$$

- **L** represents the Binary Cross-Entropy Loss function
- **y** is the true binary label (0 or 1)
- **f(x)** is the predicted probability of the positive class (between 0 and 1)

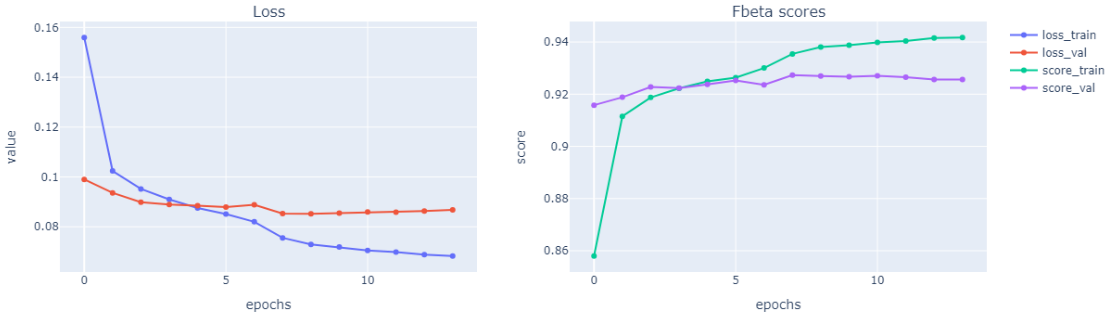


Figure 3.3: Loss value and Fbeta scores of ResNet-18 model

Figure 3.3 shows on the left graph the evolvement of loss through the training stage on both the training and the validation datasets. In the first few epochs, the loss for validation and training decreases, thus improving the model's performance. After 5 epochs, the model seems to start overfitting on the training dataset.

On the image's right side, the Fbeta-score graph is shown. The F-beta score balances precision and recall, reaching its best value at 1 and worst at 0 (Scikit-learn, 2024). It is also used to measure the model's performance. Because the classification threshold is randomly chosen for training the model (0.2), it is just helpful as a side indication of the performance. It shows the same pattern as the Loss graph, where after around 5 epochs, the validation does not improve any more, and the model seems to overfit the training dataset.

The mathematical formula for the Fbeta-score is as follows:

$$\begin{aligned} Fbeta &= \frac{(1+\beta^2) * Precision * Recall}{\beta^2 * Precision + Recall} \\ Precision &= \frac{TP}{TP+FP} \\ Recall &= \frac{TP}{TP+FN} \end{aligned} \tag{3.2}$$

where:

β **(Beta)** is a weighting factor that adjusts the balance between precision and recall. A smaller beta value gives more weight to precision and less to recall, whereas a larger beta value gives less weight to precision and more weight to recall (Brownlee, 2020),

Recall the proportion of true positives (TP) among all actual positives (TP + FN),

Precission is the proportion of true positives (TP) among all predicted positives (TP + FP).

For the calculation of the Fbeta-score for this model, a beta of 2 is chosen, meaning recall is more important than precision (Brownlee, 2020). False negatives are more important to minimize, but false positives are still important. The individual Fbeta-scores are shown in Figure 3.4.

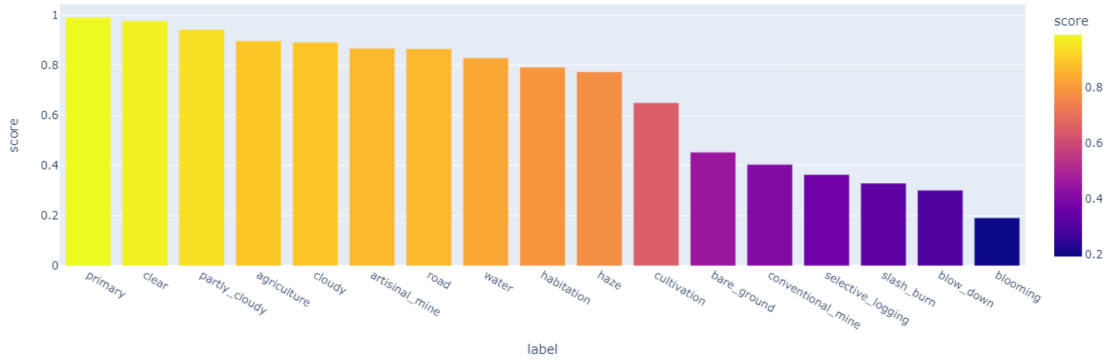
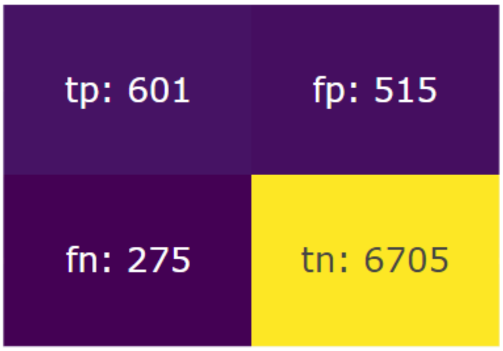


Figure 3.4: Fbeta-scores per label

The classification threshold is optimized for each class before calculating the individual Fbeta-scores. The cultivation label has a relatively low score, but the score is still high enough for the model to be useful. Therefore, checking the confusion matrix in Figure 3.5 is important. Due to the chosen beta-value, the model has a relatively high number of false positives (515) and a relatively low number of false negatives (275). In this research, a false positive is better than a false negative, since positives can still be checked by hand to see if the imagery indeed looks like cultivation fields.



tp: 601	fp: 515
fn: 275	tn: 6705

Figure 3.5: Confusion matrix for Cultivation, based on the training set (f=false, t=true, p=positive, n=negative)

3.1.4 Post-processing results

Once the chips are labelled, they need to be filtered. The first step is to select only the chips with the "cultivation" label while excluding those labelled as "habitation", since cultivated fields are often located farther away from inhabited areas (DEA, 1991). The other criteria are:

Elevation: The cultivation fields grow in an area with an elevation with a maximum of 1500 meters (Est, 2022). Therefore, chips with an elevation higher than 1500 meters are excluded.

Aspect: Aspect is the orientation of a slope, measured clockwise in degrees from 0 to 360, where 0 is north-facing, 90 is east-facing, 180 is south-facing, and 270 is west-facing. Cultivation primarily occurs on slopes facing north, east, and south, with a strong preference for east-facing slopes (Chadid et al., 2015; DEA, 1991). Therefore, a range between 45 and 135 degrees was chosen.

Slope: The slope gradient in cultivation areas generally ranges from 10° to 65° (Chadid et al., 2015).

3.2 Detection of potential laboratories

3.2.1 Dataset

For this task, the Coca-Paste Detection dataset (Pinto, 2022) has been selected, in which potential laboratories are labelled with bounding boxes. This dataset contains approximately 16500 image chips, each with a resolution of 3 meters per pixel and a size of 128x128 pixels. The dataset was curated to support the detection of potentially clandestine laboratories, providing labelled instances essential for training and validating machine learning models.

3.2.2 Classification procedure

This choice for a model is informed by the work of Pinto Hidalgo & Silva Centeno (2022), who applied a Faster R-CNN with a resnet-50 backbone to this dataset and achieved promising results. The Faster R-CNN is particularly well-suited for this dataset, as it is an efficient method for regional object detection, offering a balance between speed and accuracy. The model in this study is developed using the same parameters as those reported in their research.

The Faster R-CNN model is implemented in Python using the PyTorch framework (Ansel et al., 2024). The PyTorch training pipeline for the Faster R-CNN model, based on Rath (2023), allows for the use of a pre-trained model and customization of several parameters such as image size, batch size, learning rate, and training epochs. In this case, a pre-trained Faster-RCNN model with a resnet-50 backbone was used. The model was trained on the same parameters as the model by Pinto Hidalgo & Silva Centeno (2022). The images were of size 128 with a batch size of 4 over 30 epochs. A learning rate of 3.02×10−53.02×10−5, a threshold of 0.3, and a seed value of 0 were used to ensure reproducibility. Some minor modifications to the pipeline were required to ensure compatibility with specific Python and PyTorch versions. During training, the primary evaluation metric was the loss function, which helped guide model optimization.

3.2.3 Performance of the classification model

The performance of the model is optimized to minimise loss. Figure 3.6 shows the loss for each epoch. Initially, the model improves rapidly, but then it stabilizes. The model is trained not further than 30 epochs to prevent overfitting on the training dataset.

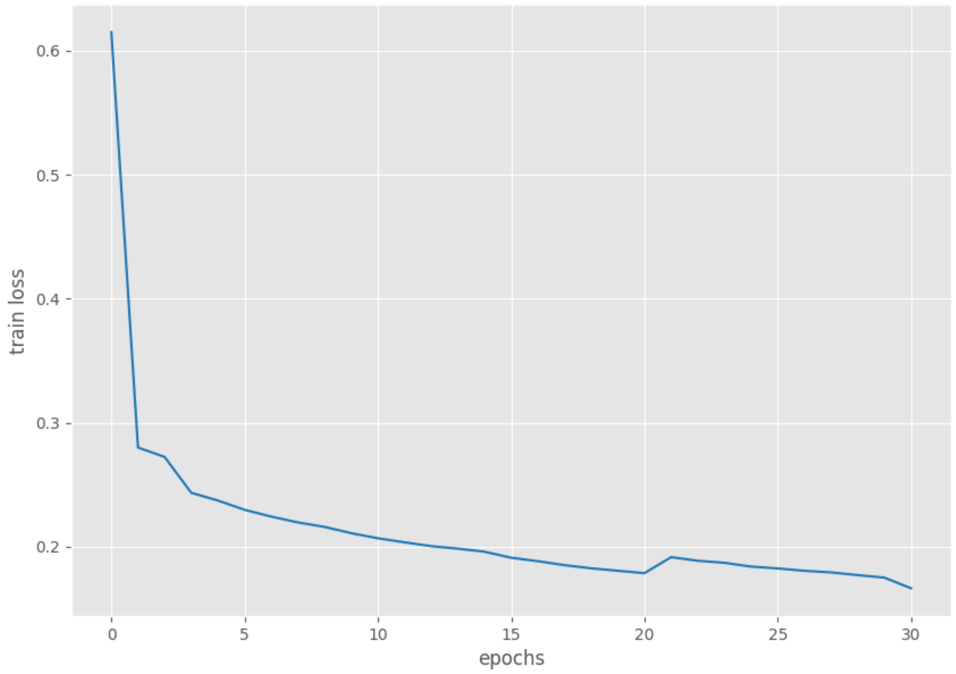


Figure 3.6: Loss value of Faster-RCNN model

Figure 3.7 shows the mean Average Precision (mAP) for each epoch. This is a different measure to evaluate the performance of an object detection model, specifically the robustness (Shah, 2022). The mAP balances precision and recall and maximizes the effect of both metrics. To find the true and false positives to calculate the mAP, the Intersection over Union (IoU) threshold is used (LightningAI, 2024). IoU measures the overlap between two boundaries, typically between the predicted boundary and the actual object boundary. The mathematical formula is as follows:

$$IoU = \frac{area\ of\ overlap}{area\ of\ union} \tag{3.3}$$

In Figure 3.7, there are two different lines for two types of thresholds: 0.5 and a range of 0.5-0.95. This range ensures that the model's performance is thoroughly evaluated without pinpointing a specific threshold. However, for this project, the precise location of the object is not as important, thus a lower threshold is accurate enough.

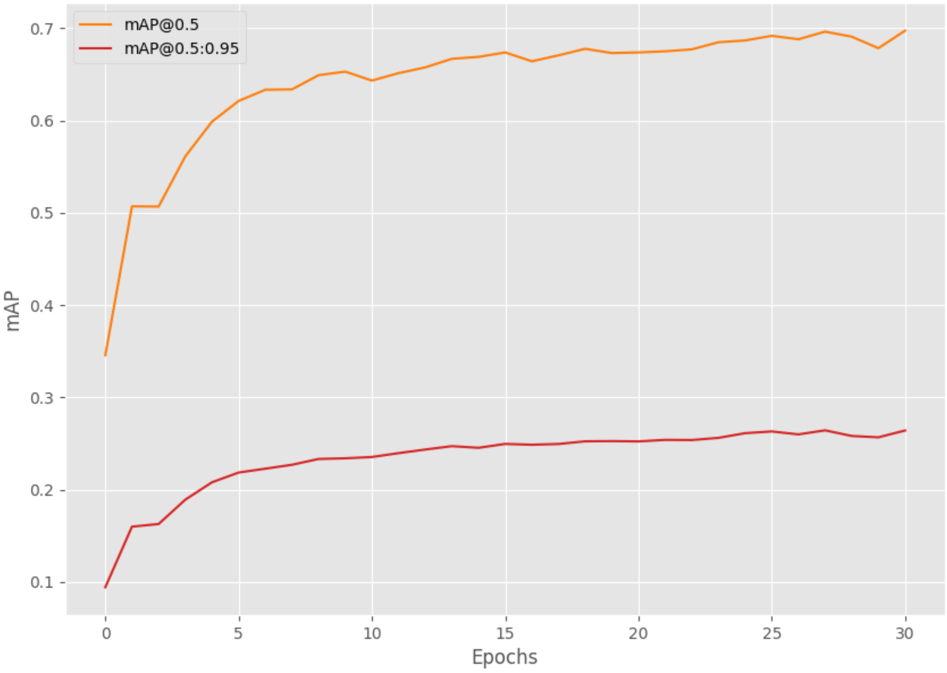


Figure 3.7: Mean Average Precision of Faster-RCNN model

The precision is defined as the number of true positives divided by the number of all detected boxes, and the recall is defined as the number of true positives divided by the number of all ground boxes (Shah, 2022). With formula 3.4, the Average Precision is calculated using precision and recall. The mAP is the mean of the AP over all classes.

$$\text{AP} = \sum_n (R_n - R_{n-1}) P_n \tag{3.4}$$

- R_n and P_n are the precision and recall at the nth threshold.

The mAP in figure 3.7 at a threshold of 0.5 is at 0.7 after 30 epochs, which is a good enough performance. Especially assuming that the mAP is a bit higher with the chosen threshold of 0.3. The model is not very robust, as the mAP is significantly lower when calculating it over a range of thresholds.

3.2.4 Post-processing results

The final product, the graph, needs points with an x and y location; therefore, the boxes need to be changed into points. The upper left corner is selected as the location of the potential laboratory. As some laboratories are very close together, laboratories in a range of 500 meters are consolidated into one laboratory.

4 | Constructing the graph

This chapter describes how the graph with the nodes and edges is constructed. The first section 4.1 focusses on attaching waterway and road edges. The second section 4.2 describes the connection of nodes through pathfinding. The third and last section describes the connecting ports and airstrips to the graph 4.3.

4.1 Attaching waterway and road edges

To model transport in rural areas, waterways, and roads must be combined to enable the possibility of crossing the water to a road on the opposite side of a waterway. Graphs from these waterways and roads are downloaded with OSMNX (Boeing, 2024), a Python package to handle Open Streetmaps (OSM) with Networkx (Hagberg et al., 2008). Filters are applied to download data from OSM. For roads, these filters are: [“highway” “motorway|trunk|primary|secondary|tertiary|unclassified”]. This means that roads with a ’residential’ label are excluded. These are roads which serve as access to housing without the function of connecting settlements. These are excluded to reduce the size of the graph and, therefore, computational time. For waterways, the following filter is applied: [“waterway” “river|canal|tidal_channel”], which are the waterways potentially navigable by boat.

The two separate road and waterway graphs need to be combined into one graph, to enable the crossing of a river and navigating the river by boat. This is done through the following algorithm illustrated in Figure 4.1. The algorithm begins with a node from the road graph. It finds the nearest edge from the water graph. Then, if the distance to the nearest water edge is smaller than the maximum distance, the water edge is split on the nearest point to the road node, a new water node is added, and a straight edge is drawn between the road and the new water node. This is done for all road nodes. Then, the algorithm repeats for all the newly added water nodes.

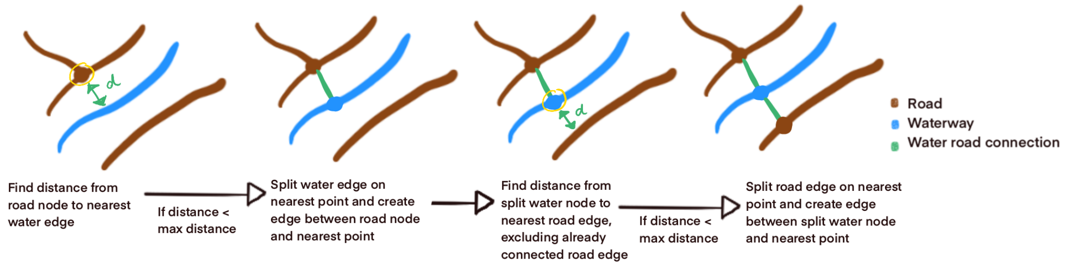

d
d
Road
Waterway
Water road connection
Find distance from road node to nearest water edge
If distance < max distance
Split water edge on nearest point and create edge between road node and nearest point
Find distance from split water node to nearest road edge, excluding already connected road edge
If distance < max distance
Split road edge on nearest point and create edge between split water node and nearest point

Figure 4.1: Concept of algorithm attaching water and road edges

4.2 Connecting nodes to graph through pathfinding

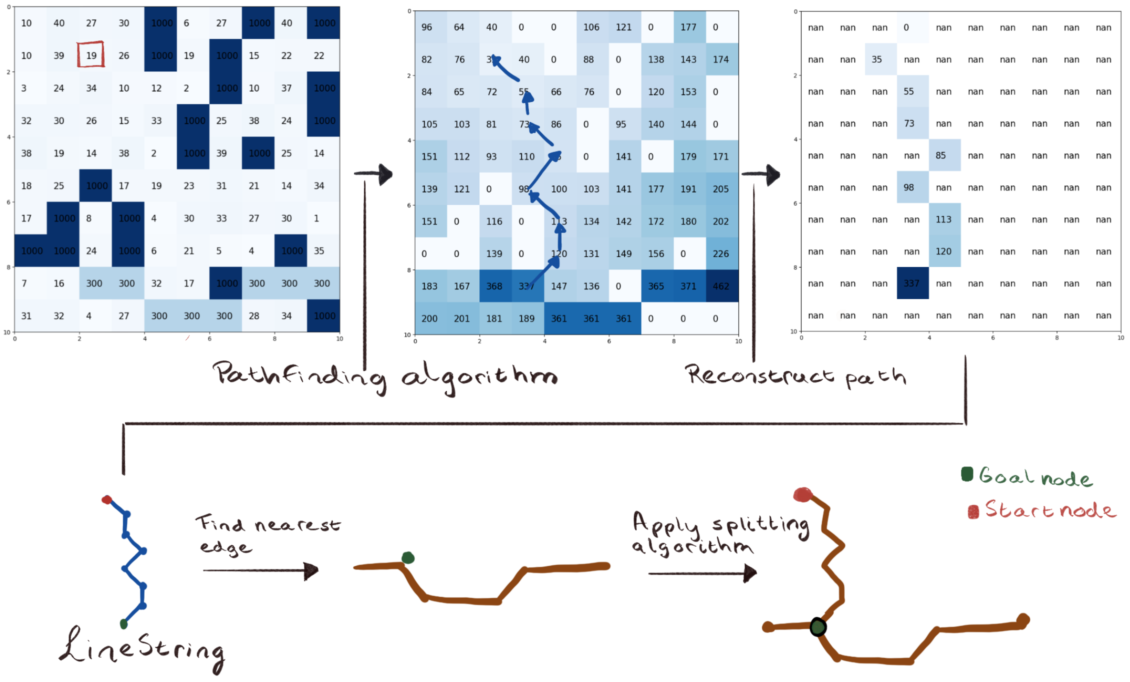
At this stage, a graph has been constructed representing the connections between waterways and roads, with the locations of potential laboratories and cultivation fields identified. To identify potentially illicit airstrips from OpenStreetMap (OSM) data, nodes were initially downloaded using the filter [“aeroway”:“runway”]. Subsequently, these nodes were refined by excluding entries sourced from ourairports.com, which catalogues airstrips associated with official airports. Additionally, airstrips with an asphalt surface were excluded from consideration.

This process resulted in three categories of nodes that remain disconnected from the graph: potential laboratories, cultivation fields, and airstrips. Integrating these nodes into the existing graph poses a challenge due to the substantial distances between these nodes and the established transportation network. Consequently, the algorithm previously employed for connecting waterways and roads, which assumes relatively short connections, is unsuitable for this task.

To address this challenge, a raster-based least-cost path approach was utilized to model realistic connections. This method accounts for terrain and generates paths of least resistance, ensuring that the resulting routes align with the landscape’s physical characteristics.

First, an elevation map with a resolution of 30 meters from Aster Digital Elevation Models (DEM) is downloaded, available through NASA’s Earth Data Explorer (NASA, 2024). This elevation data is then converted into a slope array, where areas of steepness greater than 50% are marked as inaccessible (Montagut Llauradó, 2022). The roads and waterways are incorporated into the slope array with an overlaying algorithm. The water and road graph is saved into a multidimensional array with dimensions similar to the slope array. This array is overlaid with the slope array.

This 2-dimensional array is input for the path-finding algorithm, shown in Figure 4.2. The path-finding algorithm is run between a defined start node, such as a laboratory or airstrip. It stops once it reaches a point in the raster containing a value indicating a road or waterway. This will be the end node. Too steep areas are defined with a barrier value of 1000 and road and waterways are assigned the value of 300.



10	40	27	30	1000	6	27	1000	40	1000
10	39	19	26	1000	19	1000	15	22	22
3	24	34	10	12	2	1000	10	37	1000
32	30	26	15	33	1000	25	38	24	1000
38	19	14	38	2	1000	39	1000	25	14
18	25	1000	17	19	23	31	21	14	34
17	1000	8	1000	4	30	33	27	30	1
1000	1000	24	1000	6	21	5	4	1000	35
7	16	300	300	32	17	1000	300	300	300
31	32	4	27	300	300	300	28	34	1000

96	64	40	0	0	106	121	0	177	0
82	76	35	40	0	88	0	138	143	174
84	65	72	55	66	76	0	120	153	0
105	103	81	73	86	0	95	140	144	0
151	112	93	110	85	0	141	0	179	171
139	121	0	98	100	103	141	177	191	205
151	0	116	0	113	134	142	172	180	202
0	0	139	0	120	131	149	156	0	226
183	167	368	337	147	136	0	365	371	462
200	201	181	189	361	361	361	0	0	0

nan	nan	nan	0	nan	nan	nan	nan	nan	nan
nan	nan	35	nan	nan	nan	nan	nan	nan	nan
nan	nan	nan	55	nan	nan	nan	nan	nan	nan
nan	nan	nan	73	nan	nan	nan	nan	nan	nan
nan	nan	nan	nan	85	nan	nan	nan	nan	nan
nan	nan	nan	98	nan	nan	nan	nan	nan	nan
nan	nan	nan	nan	113	nan	nan	nan	nan	nan
nan	nan	nan	nan	120	nan	nan	nan	nan	nan
nan	nan	nan	337	nan	nan	nan	nan	nan	nan
nan	nan	nan	nan	nan	nan	nan	nan	nan	nan

Figure 4.2: Concept of adding nodes to graph with path-finding algorithm

The pathfinding algorithm is based on an A* algorithm with the code from the Python XRspatial package (makepath, 2024); however, there is no predefined goal, so some adaptations were made. This results in a less efficient algorithm because it does not consider whether we move closer to a goal.

Algorithm 1: Path-finding

```
Initialize six 2-dimensional arrays of the same size:
  • Slope data: This array contains the slope values, the goal values and the
    barrier values
  • Cost: This array is initialized as an array with only zeroes and is used to
    save the costs
  • Parentx, Parenty: These arrays save the x and y location of the
    parent/previous "node" for the current cost value
  • Open, Closed: These arrays are initialized as zeroes and are used to track
    which locations should be/have been visited by the algorithm

Assign the starting location (x, y) a 1 in the Open array
Count=0
while sum of open array > 0 do
    Count+=1
    Find open location with lowest value in cost array, this is the current
    node
    Remove current node from open array, add it to closed array
    Remove current node from open array, add it to closed array
    if Count=40000 then
        At least a range of 3 km has been searched and there is no goal found
        Return
    if current node slope data = goal value then
        Backtrack from current node to get path with Parentx and Parenty arrays
        The path is saved into an array
        Return
    for each of the 8 adjacent squares, the neighbours do
        if neighbour cost = barrier value or neighbour is closed then
            Ignore neighbour
        if neighbour is horizontal or vertical then
            Cost neighbour = cost to current node +
            $(1/2) * slope_{neighbour} + (1/2) * slope_{currentnode}$
        else if neighbour is diagonal then
            Cost neighbour = cost to current node +
            $(\sqrt{2})/2) * slope_{neighbour} + (\sqrt{(2)}/2) * slope_{currentnode}$
        if Neighbour is not on open list then
            Add it to open list
            Add x,y of current square to the Parentx and Parenty of the neighbour
            Add cost of neighbour to cost array
    else if Neighbour is on open list then
        if Cost of neighbour already in cost array > cost neighbour current node
        then
            Add x,y of current square to the Parentx and Parenty of the neighbour

            Add cost of neighbour to cost array
        else
            Ignore neighbour
```

As shown in Figure 4.2, the array containing the path must be transformed into a geometry using a LineString to be able to add it to the graph. This is done with Algorithm 2, which loops through all the locations in the output array of the path-finding algorithm, similar to the last array in Figure 4.2. It creates a LineString geometry with all the coordinates in the correct order.

Algorithm 2: Transform path into geometry

```
1. list of all the $index_x$ and $index_y$ that have a value in the array = coords
 list
2. start node is current node
while len coords list > 0 do
    Remove current node of coords list
    Find x and y of current node in array with index values
    Add Point(x,y) to LineString
    Make list of $index_x$ and $index_y$ values of neighbours
    Find neighbour with $index_x$ and $index_y$ in coords list
    Set neighbour as current node
```

Then, the nearest edge to the goal node in the graph is found, and with the splitting algorithm, as mentioned in the previous section, the new road is added and connected to the graph. with waterways and roads.

4.3 Connecting ports and airstrips to graph

Potential illicit airstrips are imported from Open Street Maps (OSM) and connected to the graph using the path-finding algorithm. These airstrips are then linked with each other using straight lines (flight connections) to form a fully connected network. Ports are imported from a UNCECE database (United Nations Economic Commission for Europe, 2024) and connected to the nearest edge with a splitting algorithm. It is assumed that the path-finding algorithm is unnecessary here, as ports should be reachable through official roads. Actual shipment data is used to see which ports have connections to link the ports. This data is obtained from schedules of the shipping companies Maersk, MSC and CMA-CGM.

5 | Approach for identifying routes

5.1 Trade-offs in illicit supply chains

In criminal supply chains, traffickers face a trade-off in choosing routes (Klaassen, 2021). This trade-off is between minimizing risk, reducing cost, and optimizing transportation time. These factors often conflict, requiring traffickers to make decisions that balance their priorities. Minimizing transportation time is critical for maintaining supply chain efficiency, meeting demand, and increasing turnover rates. Reducing costs is significant as well, as it directly impacts profit margins. Meanwhile, minimizing risk is paramount to ensure that the majority of illicit goods successfully reach their intended destinations. When estimating routes within such supply chains, it is essential to account for all three factors—time, cost, and risk— to produce realistic route predictions.

5.2 Conceptualisation of routes and weights

A smuggling route in the illicit supply chain in South America is simplified for this research, see Figure 5.1. It originates at a cultivation field, from which the produce is transported to a laboratory and subsequently to a port for overseas shipment. The modes of transportation used on roads, which include the paths used to connect the added nodes, are a donkey and a truck. A motorboat is utilized for inland waterways, while maritime transportation is conducted via ship. A small aircraft is employed for transport along air routes.

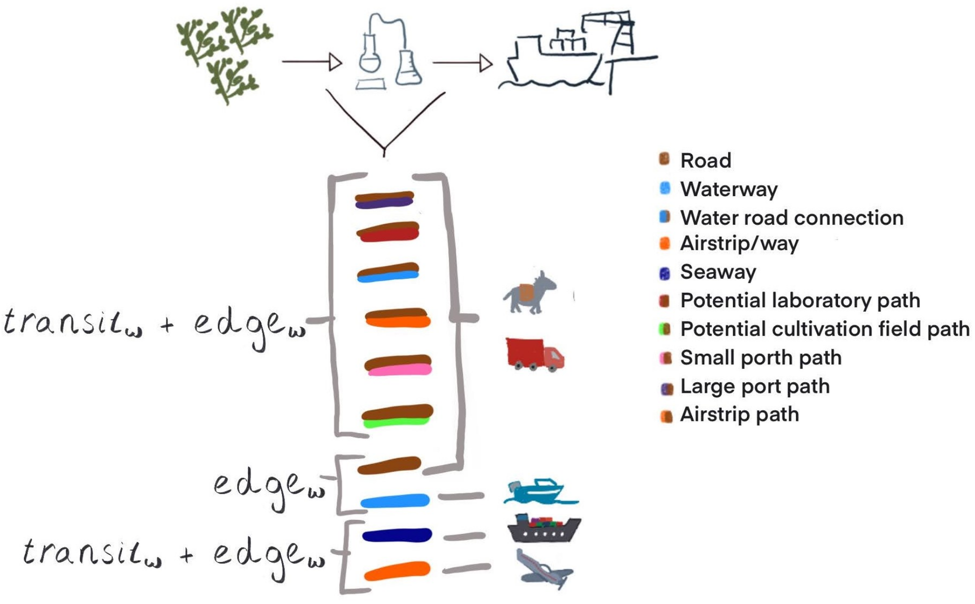


Figure 5.1: Conceptualisation of edge weights per modality and edge type

The route finding aims to predict general route and link flows in a network. Various approaches exist for predicting route choices, including deterministic, stochastic, and probabilistic methods (Prato, 2009). A deterministic approach is chosen for this proof of concept due to its simplicity and suitability for weighted graphs. Specifically, Dijkstra's shortest path algorithm is employed to identify optimal routes within the network. This also enables the comparison of trade-offs through different edge weights.

The implementation uses the NetworkX library in Python, which provides a built-in Dijkstra's shortest path function. This algorithm will be applied on three different weight parameters separately: travel time, cost, and risk. Initially, it will be executed for each origin-destination (OD) pair between potential cultivation fields and potential laboratory locations. Subsequently, laboratories with the shortest paths for each weight from the fields will be selected. Finally, the algorithm will be applied to the OD set between these selected laboratories and a subset of international ports.

The three weights—travel time, cost, and risk—are calculated separately but similarly. As illustrated in Figure 5.1 roads and waterways are assigned weights based solely on the specific edge and the modality utilized, denoted as $edge_w$. The modality is important for determining the $edge_w$. Roads with a gradient exceeding 30 degrees are assigned donkey as the transport mode, while those with a gradient of 30 degrees or fewer are assigned truck. Connections involving roads include an extra weight to account for transit costs, represented as $transit_w$. This approach is also applied to seaways and airways, where transit-related weights are incorporated alongside the edge-specific weights.

5.3 Edge weight equations

This section describes the equations used to quantify $edge_w$ and $transit_w$ for the three weights, travel time, cost, and risk.

5.3.1 Travel time

The travel time for a given edge, denoted as $edge_w$, represents the time required to travel a given edge, calculated based on the edge length and the speed associated with the transport modality. Transit time, $transit_w$, accounts for the estimated duration needed to transfer products between transportation modes. This leads to the following formulas for calculating the travel time of an edge:

$$\begin{aligned} edge_w &= length_e / speed_m \\ traveltime_e &= edge_w + transit_w * transit_{bool} \end{aligned}$$

where:

$speed_m$ is the speed of the modality in km/h,

$lengt_e$ is the length of an edge in km,

$edge_w$ represents the time spent travelling in hours,

$transit_w$ represents the time needed to transfer products between transportation modes in hours,

$transit_{bool}$ is a 1 or 0, depending on whether the edge has a transit time,

$traveltime_e$ is the combination of time spent travelling and time spent in transit (hours).

In Table 5.1, the values for calculating $edge_w$ that depend on a modality are shown. If the graph already contains a value for speed from OSM, then that speed is chosen, otherwise the values from the table are used.

Transport Mode	Speed (km/h)	Source
Truck	41	(Wilches et al., 2020)
Donkey	5	(Varga et al., 2016)
Motorboat	45	(BoatDriving.org, 2024)
Plane	357	(Taylor, 1982)
Ship	23.73	(Agarwal, 2024)

Table 5.1: Speeds of various transport modalities.

Table 5.2 lists the estimated values for $transit_w$. The transit times for ports and airstrips are divided over the connection road to the airstrip or port and the airway or seaway. This is done to account for the possibility of transit from one aircraft or ship to another. Based on United Nations Conference on Trade and Development (2023), it is assumed that the port transit time, the time it takes for a container to arrive until the ship leaves, is approximately 24 hours and does not differ for a small or large port.

Path Type	Transit Time (hours)
Water road connection	0.5
Small port path	12
Large port path	12
Airstrip path	0.75
Seaway	24
Airway	1.5

Table 5.2: Transit times for various edge types

5.3.2 Cost

Cost estimation is influenced by multiple factors. The primary costs are labour, fuel, bribes, and vehicle purchase and maintenance. Given the challenges in converting purchase and maintenance costs into a per-edge cost, this factor is excluded from the analysis. The $edge_w$ represents the cost spent for transport, calculated based on fuel consumption and wage for labour. Additionally, the transit, $transit_w$, accounts for bribes and taxes. Note that bribes are included, since we model the transport of illegal goods. Each vehicle type has a specific carrying capacity, and for standardization purposes, it is assumed that each edge involves the transport of 500 kg of goods. This may require multiple vehicle trips per edge. Although the product's weight varies across different supply chain stages, this variation is not further considered in the cost calculations. This leads to the following sets of equations:

$$\begin{aligned} hours_{mo} &= hours_w * 52/12 \\ wage_h &= wage_{mo}/hours_{mo} \\ wage_{km} &= wage_h/speed_m \end{aligned}$$

where:

$hours_w, hours_{mo}$ are the working hours per week and month,

$wage_h, wage_{mo}$ is the wage in € per hour and per month,

$speed_m$ is the speed of the modality in km/hour,

$wage_{km}$ is the wage in € per km travelled.

and

$$\begin{aligned} edge_w &= fc * length_e + wage_{km} * length_e \\ transit_w &= tax_e * load_m \\ cost_e &= (edge_w + transit_w * transit_{bool}) * 500/load_m \end{aligned}$$

where:

fc_m is the fuel consumption of a modality in €/km,

$length_e$ is the length of an edge in km,

$edge_w$ represents the cost of travelling in €,

tax_e is an estimation of the taxes and bribe money paid in transit in €/kg,

$load_m$ represents the load a transportation mode can carry in kg,

$transit_w$ represents cost of bribes and taxes in €,

$transit_{bool}$ is a 1 or 0, depending on whether the edge has a transit time,

$cost_e$ is the combination of cost spent travelling and cost spent in transit in €.

Table 5.1 shows the values for calculating $edge_w$ that depend on a modality, fuel consumption and wages. The average wage is taken for all the modalities except for the aircraft, as it needs an educated pilot (Gallimore, 2024). Wages and fuel consumption are assumed to be 0 for ships, as the illicit products are often illegally hidden between licit products in containers paid for by licit companies.

Transport	**Fuel consumption (€/km)**	**Wage (€/month)**	**Source**
Truck	0.11	1020	(Prices, 2024)
Donkey	0	1020	
Motorboat	0.22	1020	(Shafran, 2024)
Plane	0.24	1700	(Stickney, 2024)
Ship	0	0	

Table 5.3: Fuel consumption and wage for various transport modalities

Table 5.4 shows the estimated bribing costs per kilogram. The costs are based on assumptions and an expert interview, as there is no official data available on the bribing costs. At an illicit airstrip and small port, they are assumed to be €50. Transitions from a truck to a boat are assumed to have a lower cost of €25, as there is less risk and expertise involved. A larger port is assumed to cost more, as there is more risk involved for the bribed individuals: €75. Similar to transit times, the transit costs for ports and airstrips are divided over the edges to account for the possibility of transit from one aircraft or ship to another.

Transit Type	**Transit Cost (€/kg)**
Water road	25
Small port	25
Large port	50
Airstrip path	25
Seaway	50
Airway	50

Table 5.4: Transit costs for various edge types

5.3.3 Risk

Risk is conceptualised as the probability of interception and the subsequent loss of valuable goods and prosecution of the people involved. Multiple factors influence this risk, such as the frequency of police patrols along a route, the terrain, and the detectability of the smuggling method. This analysis considers two specific risk factors: the detectability of the modality and the risk of being caught during transit. Each modality and transit type is assigned a value between 0 and 1 to represent relative risk. Although edge length is not factored into the risk calculations, potentially favouring longer over shorter routes may reflect real-world conditions where shorter routes with frequent crossings are busier and thus pose a higher risk of interception.

However, this simplification does not account for the increased risk associated with prolonged travel times:

$$risk_e = edge_w + transit_w * transit_{bool}$$

where:

$edge_w$ represents the risk of travelling on the edge,

$transit_w$ represents the risk of transit,

$transit_{bool}$ is a 1 or 0, depending on whether the edge has a transit time,

$risk_e$ is the combination of the travelling risk and the transit risk.

Table 5.5 presents the estimated risk values associated with travelling along an edge for each mode of transport. Table 5.6 outlines the risks associated with the transfer of goods to another modality. Due to the lack of data to assign absolute risk values, relative values have been inferred based on assumptions grounded in the visibility and detectability of each modality and transit type. Transit risks are generally estimated to be higher than travel risks because transferring goods between modalities is a more noticeable process, increasing the likelihood of detection.

Air travel is assigned the highest risk value for transport modalities. Aircraft, particularly unregistered flights, are highly visible both to radar systems and from the ground, making them more susceptible to detection. In contrast, donkeys are considered the least risky modality, as they are typically employed in remote, mountainous regions where the likelihood of detection is minimal. Trucks and motorboats are assumed to have comparable risk levels, as they can blend into general traffic or maritime activity. Still, they are more vulnerable to detection because they operate in less isolated areas with a greater law enforcement presence. Ships are considered to pose the lowest travel risk, as containers aboard a ship are rarely inspected once loaded.

Transport	**Risk**
Truck	0.02
Donkey	0.01
Motorboat	0.02
Plane	0.1
Ship	0.05

Table 5.5: Risk values for various transport modalities

For transit activities, the transfer of goods to motorboats is estimated to involve the least risk, given the typically rapid and discreet nature of such operations. Transit in small ports or airstrips is assigned a moderate risk level, as the process requires more time and increases the potential detection window. Large ports are considered the highest-risk transit locations due to their advanced technological screening systems and the presence of law enforcement dedicated to monitoring illicit activity.

Transit Type	**Risk**
Water road connection	0.3
Small port path	0.25
Large port path	0.55
Airstrip path	0.25
Seaway	0.5
Airway	0.5

Table 5.6: Risk values for various edge types

6 | Case study

This chapter describes the results of a case study in a coastal region between Colombia and Venezuela. Section 6.1 gives a more detailed description of the region. The second section 6.2 describes the identification of illicit objects with machine learning. The third section 6.3 describes the construction of the graph. The fourth and last section 6.4 describes the identification of routes.

6.1 Area of the case study

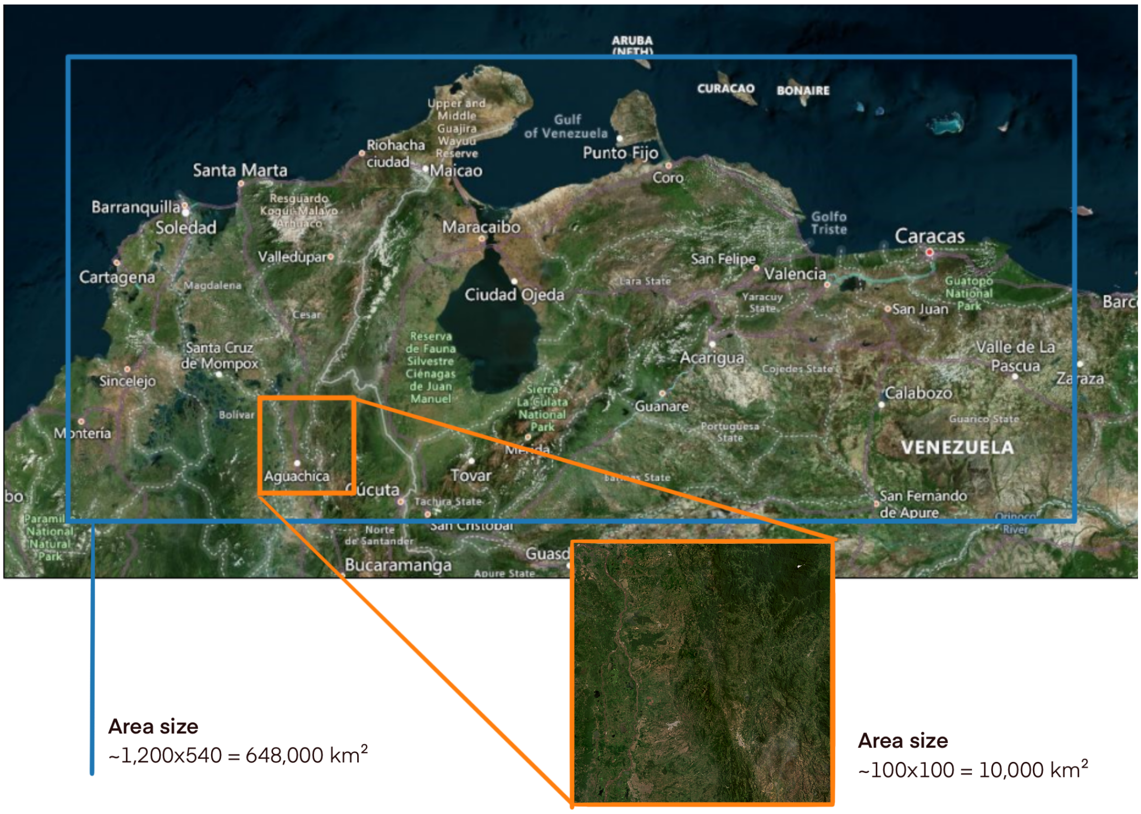


Figure 6.1: Area of the case study

Figure 6.1 shows the areas of this case study, which include part of the border and coastal regions between Colombia and Venezuela. Two distinct regions were defined for analysis: a larger area (the blue rectangle) and a smaller area (the orange rectangle), each designated for specific analytical purposes. The larger region is chosen for the presence of international ports in both Venezuela and Colombia and has a surface area of approximately 1200x540 km^2. The smaller area in Colombia near the Venezuelan border, with a surface area of approximately 100x100 km^2, is identified as having a higher likelihood of containing cultivation sites and laboratories associated with illicit activities, as suggested by Pinto Hidalgo & Silva Centeno (2022). Consequently, machine learning algorithms are applied exclusively to this smaller area to focus on more relevant regions and reduce computational demands. A close-up of this area is displayed in Figure 6.2.

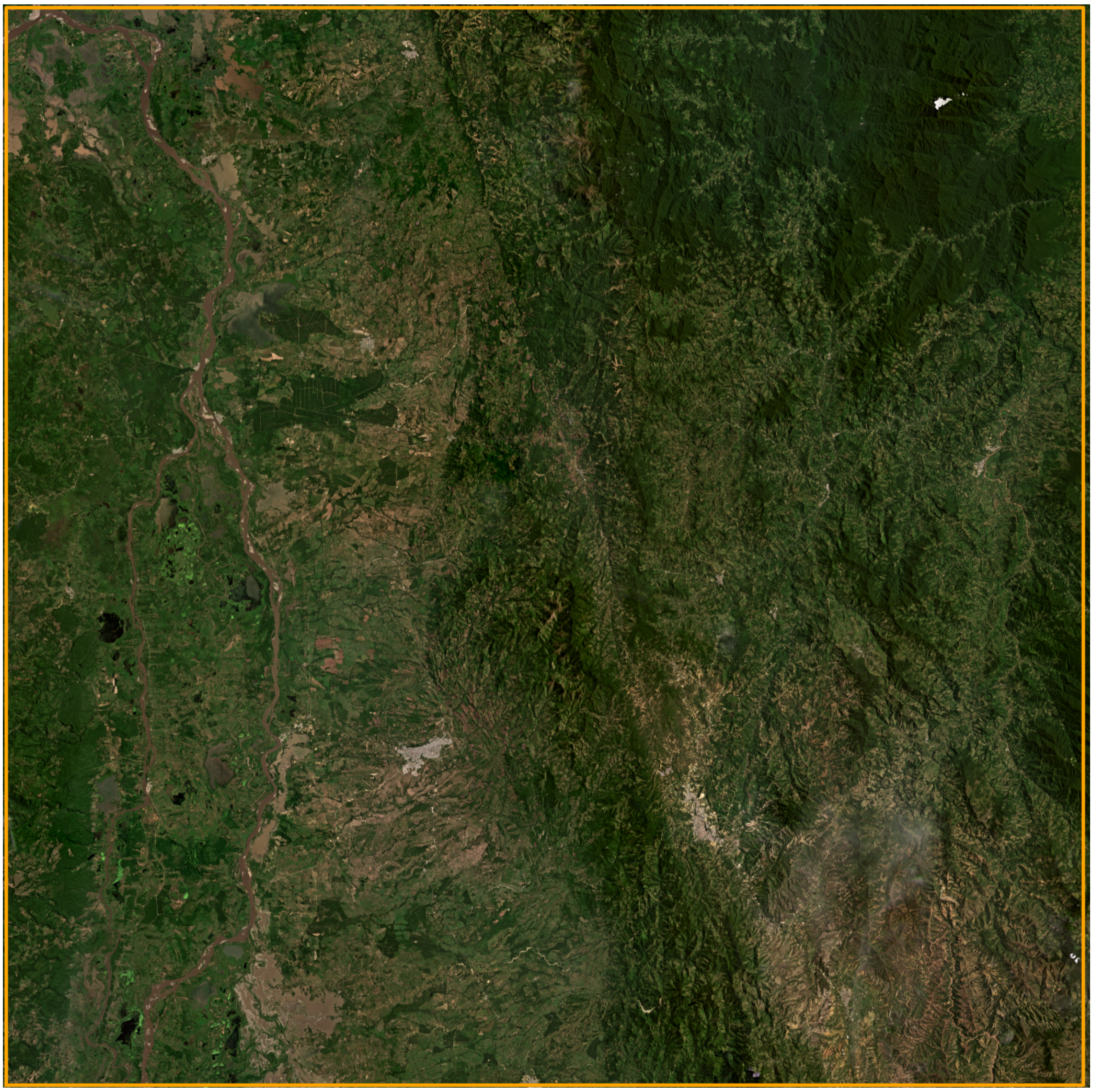


Figure 6.2: Close up of the small area. Satellite image retrieved from Planet Labs PBC (2024b).

6.2 Identifying objects with machine learning

The NICFI programme gives access to PlanetScope imagery of tropical forest regions to support research to conserve these ecosystems, with new imagery provided monthly (Planet Labs PBC, 2024a). For this case study, images from July and August were combined to create a dataset with minimal cloud cover. Each image, measuring 4096x4096 pixels at 4,77 meters ground sampling distance (GSD [1]), is used as input for machine learning models. However, to meet the models' requirements, smaller image "chips" are extracted: 128x128 pixels for identifying laboratories and 256x256 pixels for cultivation fields. Thus, the original images are clipped into chips of these specified sizes. Figure 6.3 presents a 4096x4096 image, with pink and yellow squares in the upper left corner illustrating the dimensions of these splintered chips.

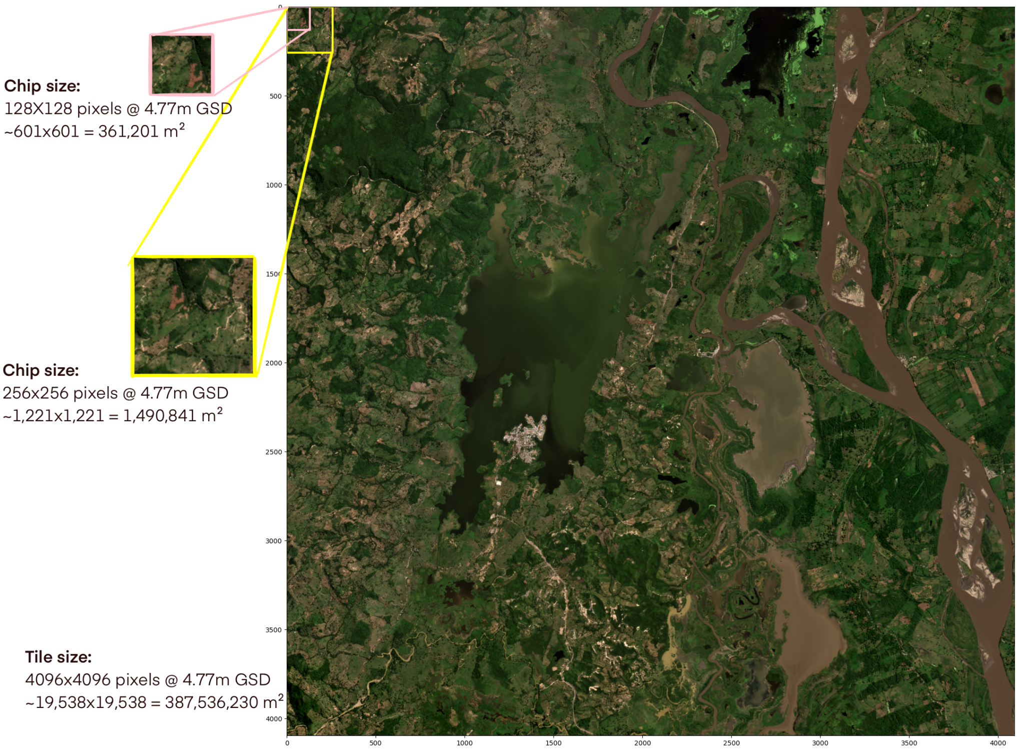

Chip size:
128X128 pixels @ 4.77m GSD
~601x601 = 361,201 m²
Chip size:
256x256 pixels @ 4.77m GSD
~1,221x1,221 = 1,490,841 m²
Tile size:
4096x4096 pixels @ 4.77m GSD
~19,538x19,538 = 387,536,230 m²
0
500
1000
1500
2000
2500
3000
3500
4000

Figure 6.3: Example of a chip of size 4096x4096, 256x256, 128x128. Satellite image retrieved from Planet Labs PBC (2024b).

Some tiles are partially outside the smaller case study area, and therefore, some chips are as well. The potential cultivation fields and laboratories identified on these chips will not be included in the graph.

6.2.1 Potentially illicit cultivation fields

The dataset, after clipping, contains 15000 chips. The distribution of predicted labels, as depicted in Figure 6.4, diverges from the distribution of labels in the dataset used for model training (Figure 3.2). This discrepancy can be attributed to the selection criteria of the satellite imagery, prioritising minimizing cloud cover. Consequently, there is a reduction in the frequency of "cloudy" and "partly cloudy" labels. Additionally, the dataset contains relatively fewer instances of the "primary" label, indicative of dense forest coverage, and higher proportions of "agriculture", "habita-

[1] GSD is the distance between pixel centres measured on the ground

tion", and "cultivation" labels, likely reflecting the geographic characteristics of the case study region.

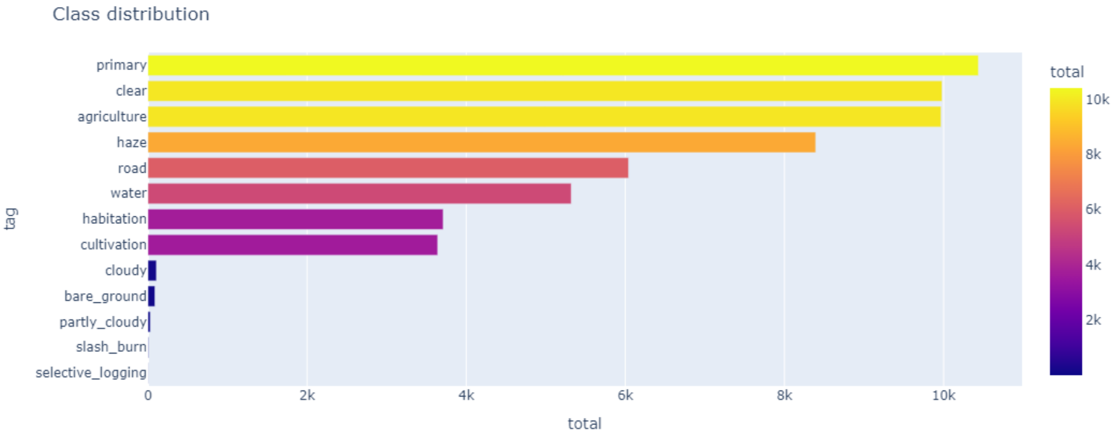


Figure 6.4: Distribution of labels.

An increase in the "haze" label appears to result from inaccuracies in the model's ability to predict haze correctly. Figure 6.5 illustrates examples of randomly selected chips labelled as "haze" that, upon inspection, do not exhibit haze. However, as the "haze" label is not critical to this research, these inaccuracies do not impact the outcomes of the case study.

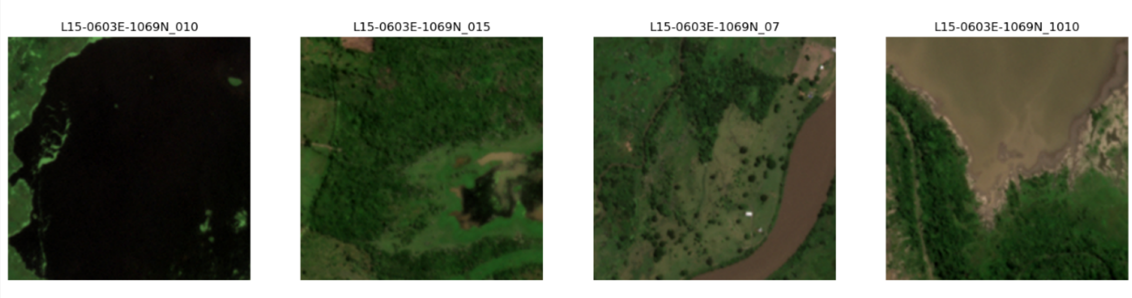


Figure 6.5: Examples of chips with haze labels. Satellite image retrieved from Planet Labs PBC (2024b).

The primary focus lies on chips classified with the “cultivation” label, which total 3,641 instances. Representative examples are shown in Figure 6.6. Although visual confirmation of cultivation presence in these chips is challenging, no instances appear to be entirely implausible as cultivation fields.

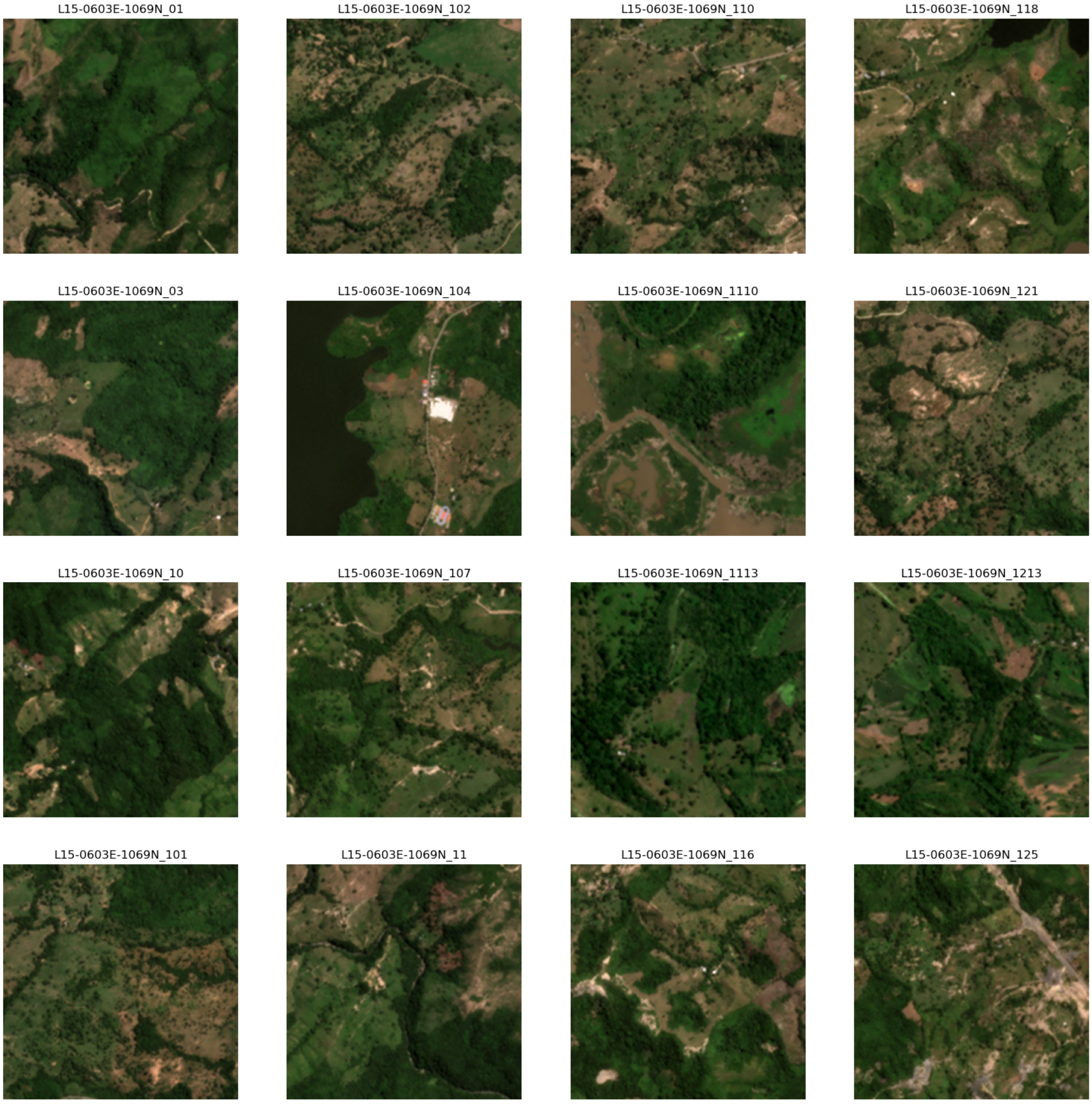


Figure 6.6: Examples of chips with cultivation labels, unfiltered (Planet Labs PBC, 2024b)

After label prediction, further filtering of the chips was performed. Initially, chips containing the “habitation” label were excluded, with results summarized in Figure 6.7. The “habitation” label may be overpredicted; for instance, chip 03 was excluded despite containing minimal habitation features.

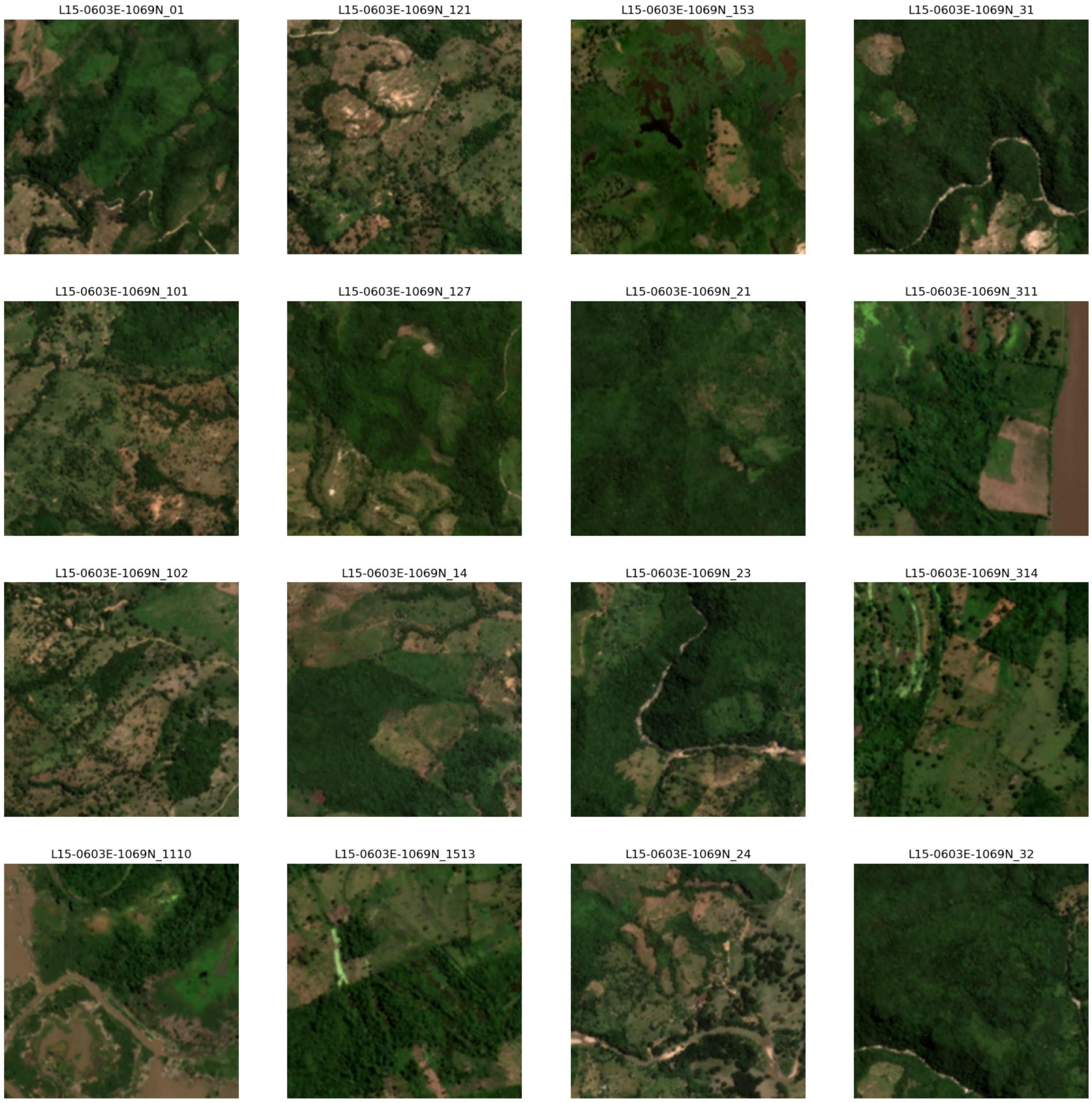


Figure 6.7: Examples of chips with cultivation labels, but no habitation labels. Satellite image retrieved from Planet Labs PBC (2024b).

Following additional filtering based on slope and aspect, 1,364 potential illicit cultivation fields were identified. Figure 6.8 provides an overview of the locations of these potential cultivation fields in the area.

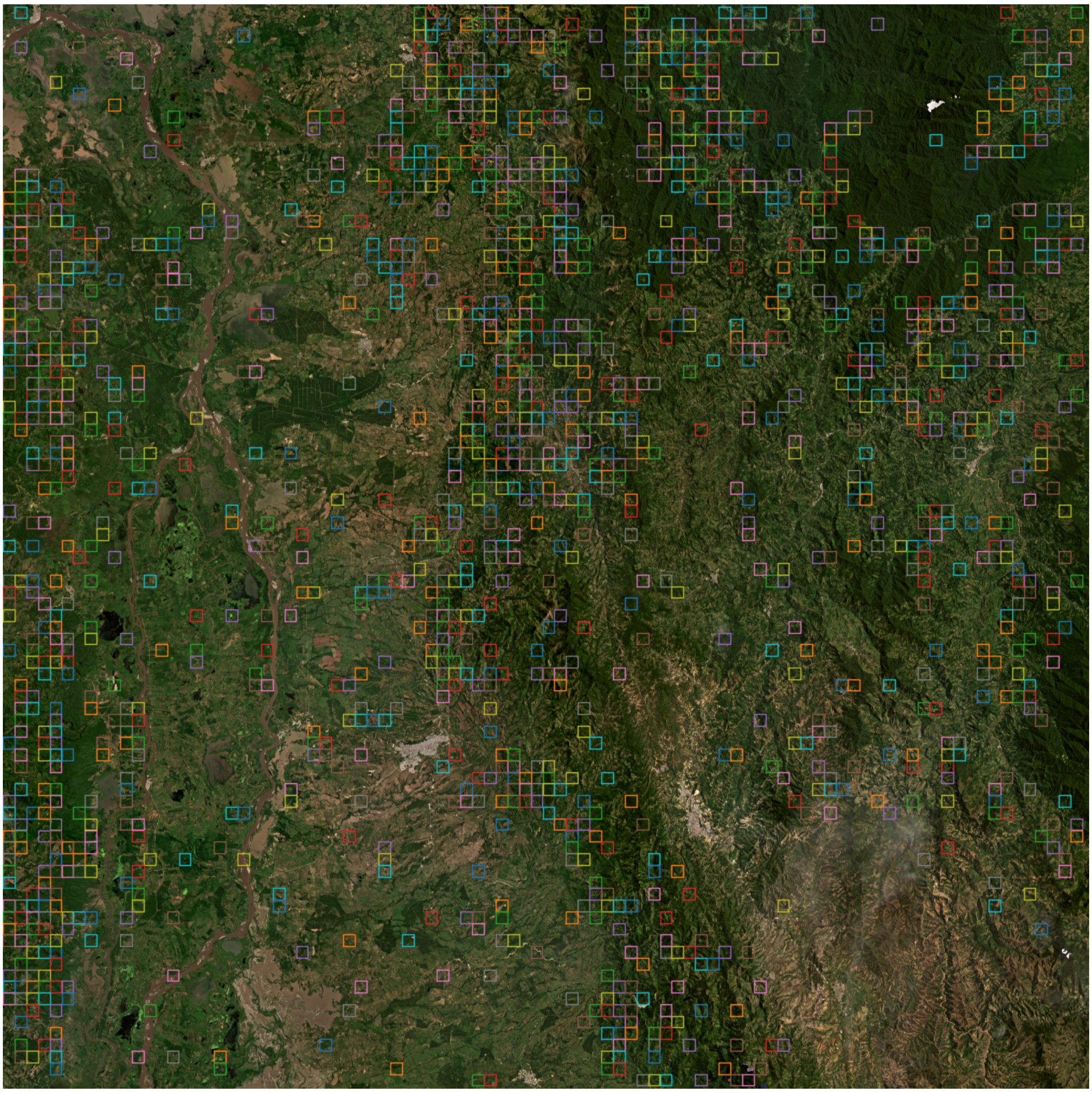


Figure 6.8: Locations of identified potential illicit cultivation fields. Image adapted from Planet Labs PBC (2024b).

6.2.2 Potentially illicit laboratories

The dataset for identifying potential illicit laboratories contains approximately 40,000 chips. Initially, 12,445 potential illicit laboratories are identified. Figure 6.9 gives examples of predicted boxes with a high confidence score that align visually with the characteristics of illicit laboratories, such as structures resembling roofed buildings located in secluded areas.

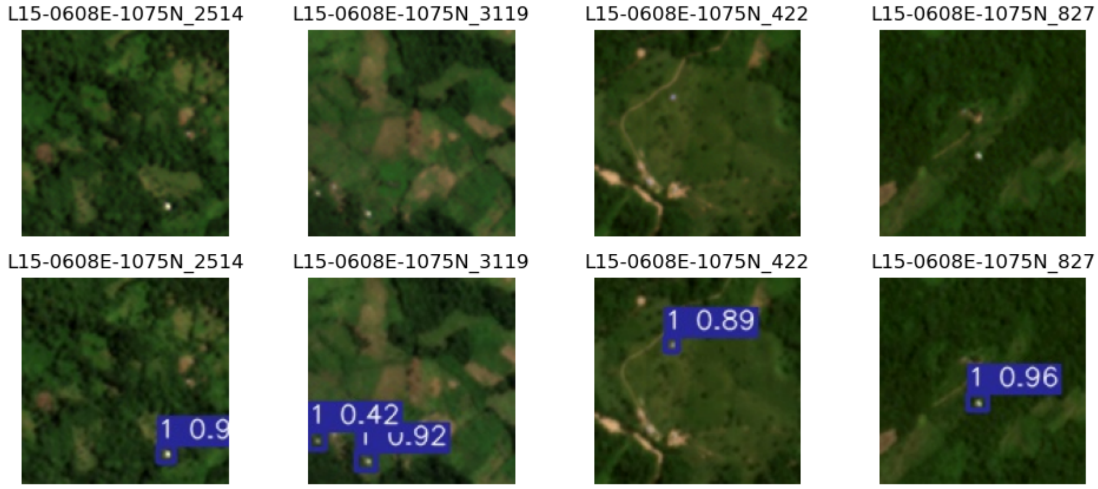


Figure 6.9: Examples of plausible laboratory predictions. Image adapted from Planet Labs PBC (2024b).

Contrarily, Figure 6.10 presents examples of predictions with lower confidence scores. These predictions are less likely to represent illicit laboratories, as they are often situated in areas with numerous other buildings. An example near a river, despite its relatively low confidence score, could still plausibly represent a potential laboratory given its proximity to water. Based on the visual and confidence score evaluation, a threshold of 0.9 was selected to retain predictions most likely to correspond to laboratories.

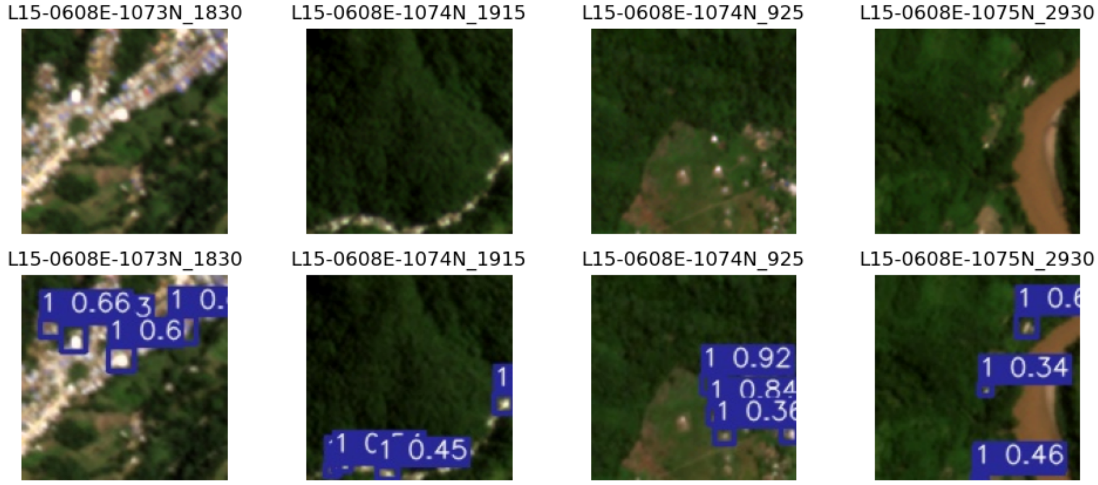


Figure 6.10: Examples of implausible laboratory predictions. Image adapted from Planet Labs PBC (2024b).

Using the threshold of 0.9 reduces the number of potential laboratories to approximately 3600 potential illicit laboratories. However, as the proximity of some predicted laboratories suggests unrealistic clustering, predictions within a 500 m radius are consolidated into single instances. The identified laboratories outside the case study area have also been removed. This results in approximately 2800 distinct laboratory predictions, of which the locations are depicted in Figure 6.11.

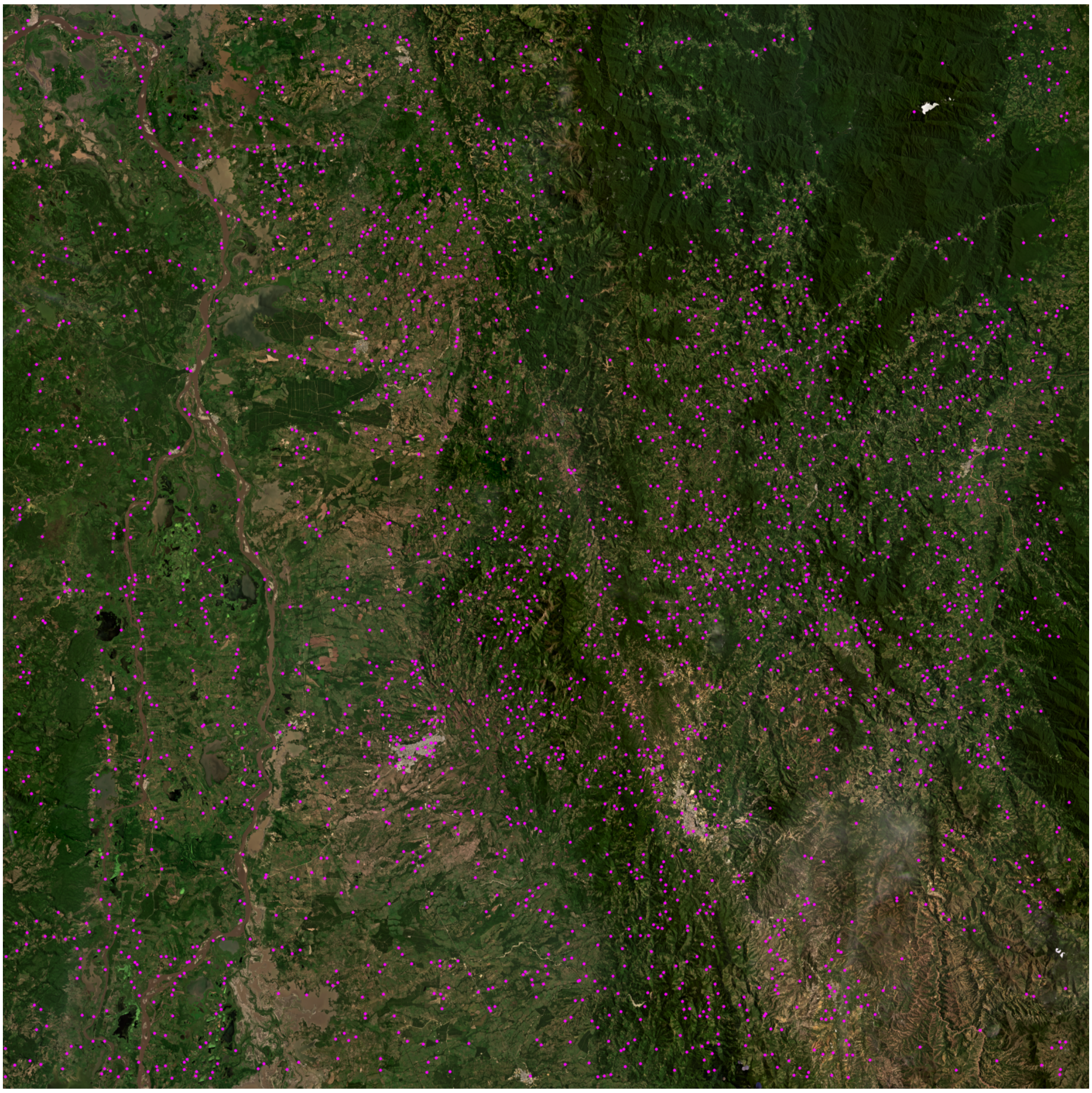


Figure 6.11: Locations of identified potential illicit laboratories. Image adapted from Planet Labs PBC (2024b).

6.3 Constructing the graph

The initial graph comprises two separate graphs: one containing the roads and the other the waterways of the large area (see the image at the bottom of Figure 6.12). These graphs are obtained from OpenStreetMap (OSM) via the OSMnx library (OpenStreetMap contributors, 2017; Boeing, 2024). The raster, shown on top of the graph, is used to improve the run time of the algorithms applied to the larger area. By segmenting the graph according to the raster sections, the number of edges required for search algorithms, such as those used to identify the nearest edge, is reduced.

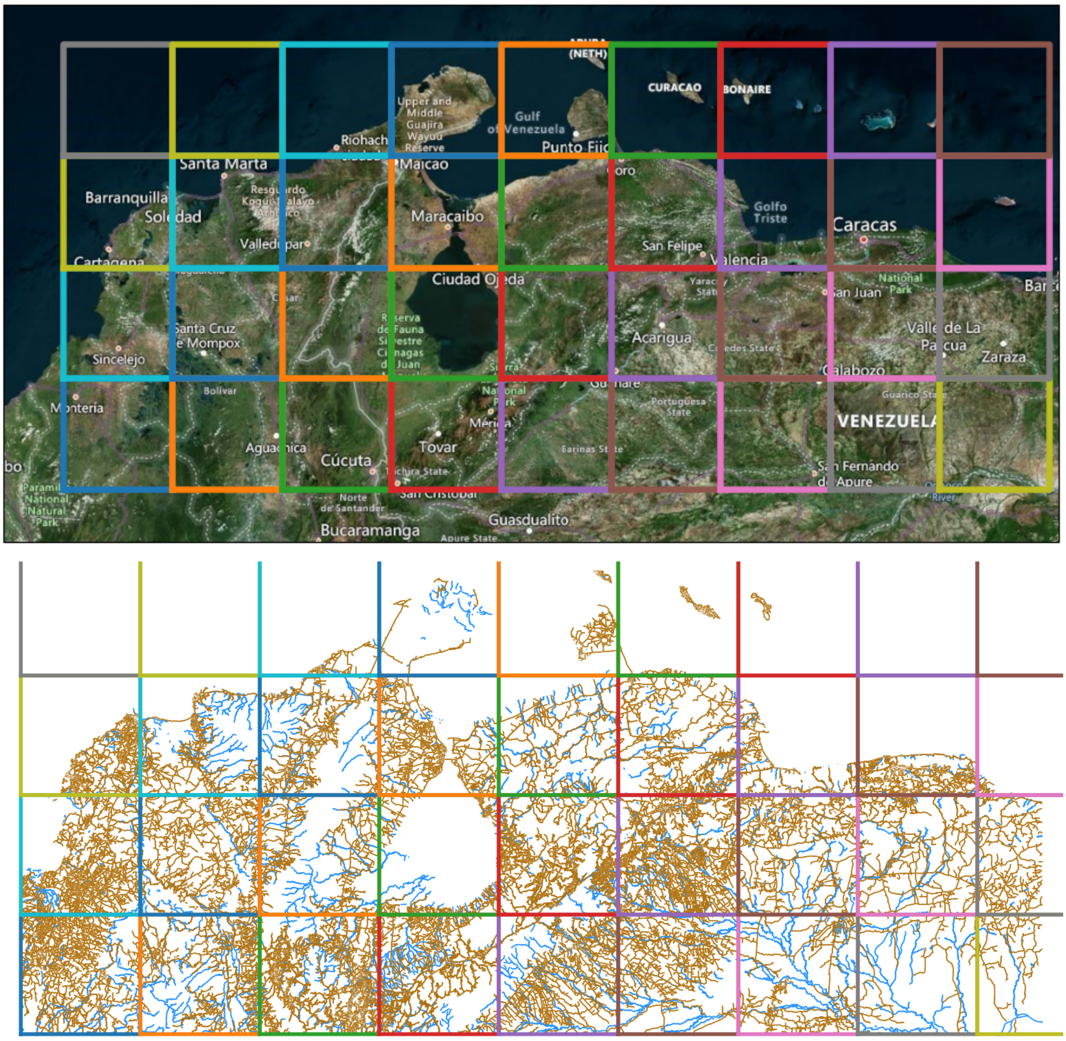


Figure 6.12: Raster to clip graph for improved computational efficiency.

6.3.1 Attaching waterway and road edges

The algorithm, described in Figure 4.1, is applied to connect the water and road graphs. Figure 6.13 shows an example of the output, where multiple connections are made between the roads and waterways. The algorithm initiates connections from the road nodes. Therefore, connections may not always align with the most visually intuitive points. For instance, a point further along a road edge might be nearer to the river yet lacks a corresponding node, leading to a less realistic connection.

Additionally, two limitations of the current implementation are noted: (1) the maximum allowable distance for connections appears to be set slightly too high, and (2) connections to the road network on the opposite side of the river are not where one

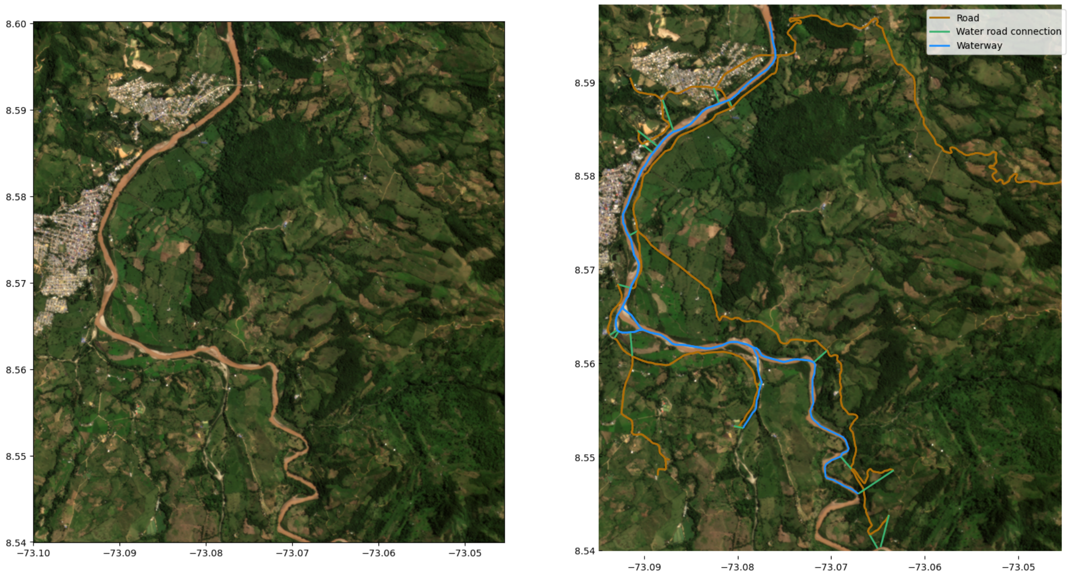


Figure 6.13: Example of connected waterway and road edges. Image adapted from Planet Labs PBC (2024b).

would visually expect them. In the example from figure 6.13 no connections are made to roads on the opposite riverbank, despite appearing to be realistic. This is because the algorithm only considers road edges that are not yet connected to water edges when establishing links across the river.

6.3.2 Connecting nodes through pathfinding

To connect the airstrips from OSM in the smaller area and the identified potential cultivation fields and laboratories to the graph, the pathfinding algorithm, outlined in Figure 4.2, is applied. Before executing this algorithm, the elevation raster must first be converted.

Figure 6.14 displays the three transformation stages of the raster required for pathfinding. The raster has a resolution of 30 meters of ground sampling distance (GSD). Initially, the raw data consists of elevation values, which are processed into a slope raster using the XRSpatial library (makepath, 2024). Then, this raster is overlaid with barrier, road, and waterway data.

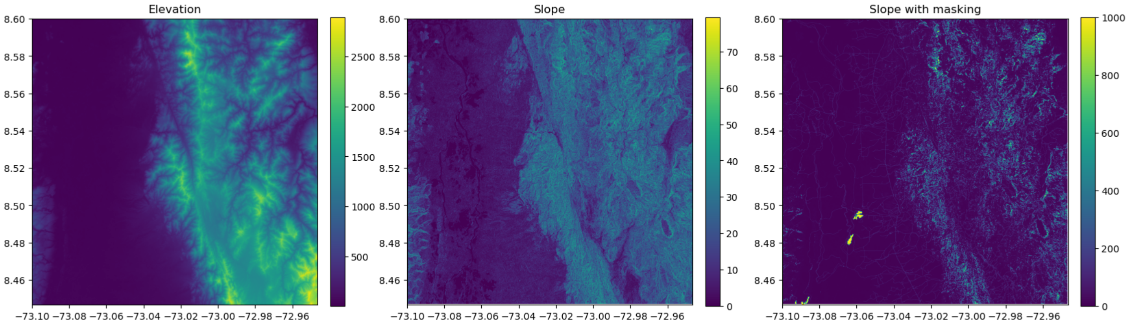


Figure 6.14: Slope raster - elevation raster - masked elevation raster. Data adapted from NASA (2024).

Pathfinding is performed with the finalized raster, resulting in edges based on slope data. An example of the final output is presented in Figure 6.15, where a field is connected to the network via a path that adheres to the least steep route, as determined by the slope raster. In the satellite image, the predicted path appears to correspond to a visible feature, which may represent either a road or a river.

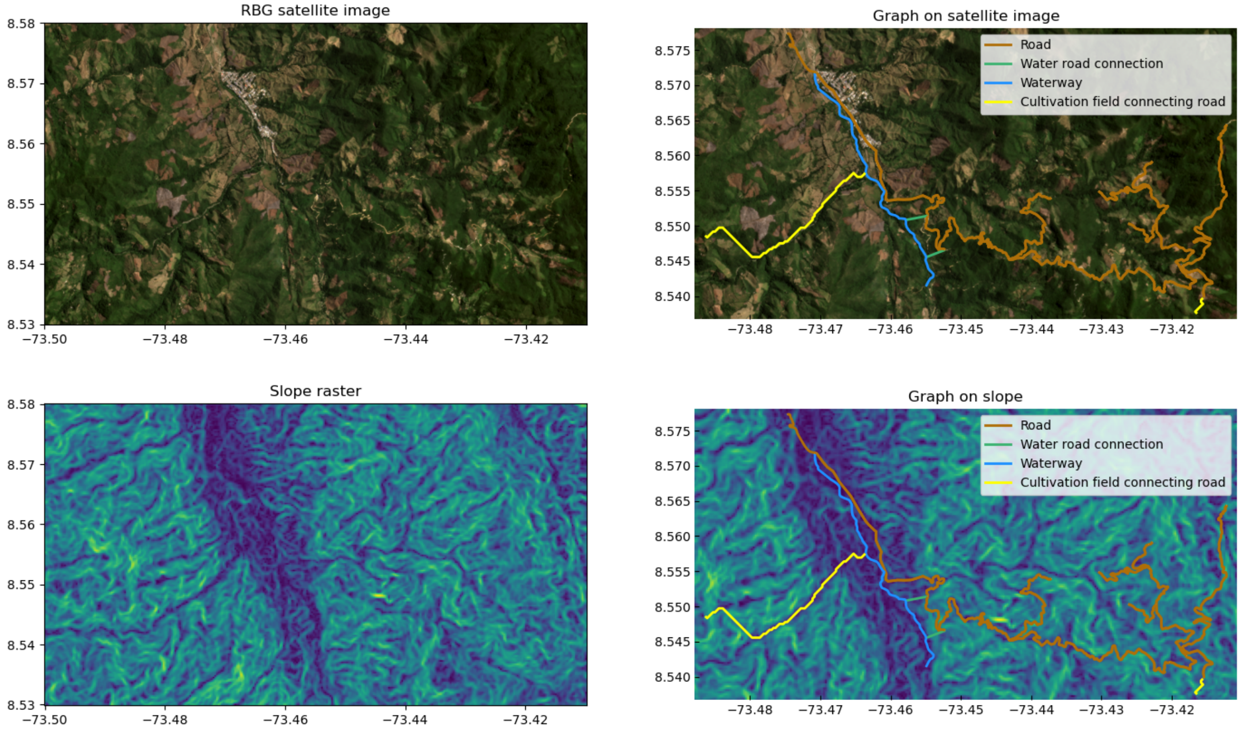


Figure 6.15: Example of a node connected to graph through pathfinding. Image adapted from Planet Labs PBC (2024b) and NASA (2024).

Figure 6.16 presents all nodes incorporated into the graph within the small area following the application of the pathfinding algorithm. These nodes include 11 airstrips, 926 fields, and 2,314 laboratories. Due to a model error related to their geometry type, four airstrips remain unconnected. This issue is not expected to significantly affect the outcomes, as other nearby airstrips provide connectivity.

The number of field and laboratory nodes added to the graph is lower than the final selection of cultivation fields and laboratories. This reduction occurred because nodes located in inaccessible areas or those lacking a viable path within a minimum range of 3 kilometres were excluded from the graph. For instance, as shown in Figure 6.16, several laboratories and cultivation fields identified in the lower-left corner of Figures 6.8 and 6.11 were not added to the graph.

Increasing the pathfinding range could enhance the number of connections, potentially improving alignment with real-world conditions. However, such an extension would require additional computational resources or improvements in algorithmic efficiency.

6.3.3 Connecting ports and airstrips

The final step involves connecting ports and airstrips within the larger area to the graph and adding connections between them. Instead of utilizing pathfinding, these nodes are linked by creating straight edges to the nearest edge in the graph. Although this approach is less precise, it offers significant computational efficiency. The resulting graph, illustrated in Figure 6.17, excludes airways, as their inclusion causes clutter. Two ports that don't have intercontinental connections are assigned to be small ports: Puerto Cabello (VEPBL) in Venezuela and Barranquilla (COBAQ) in Colombia.

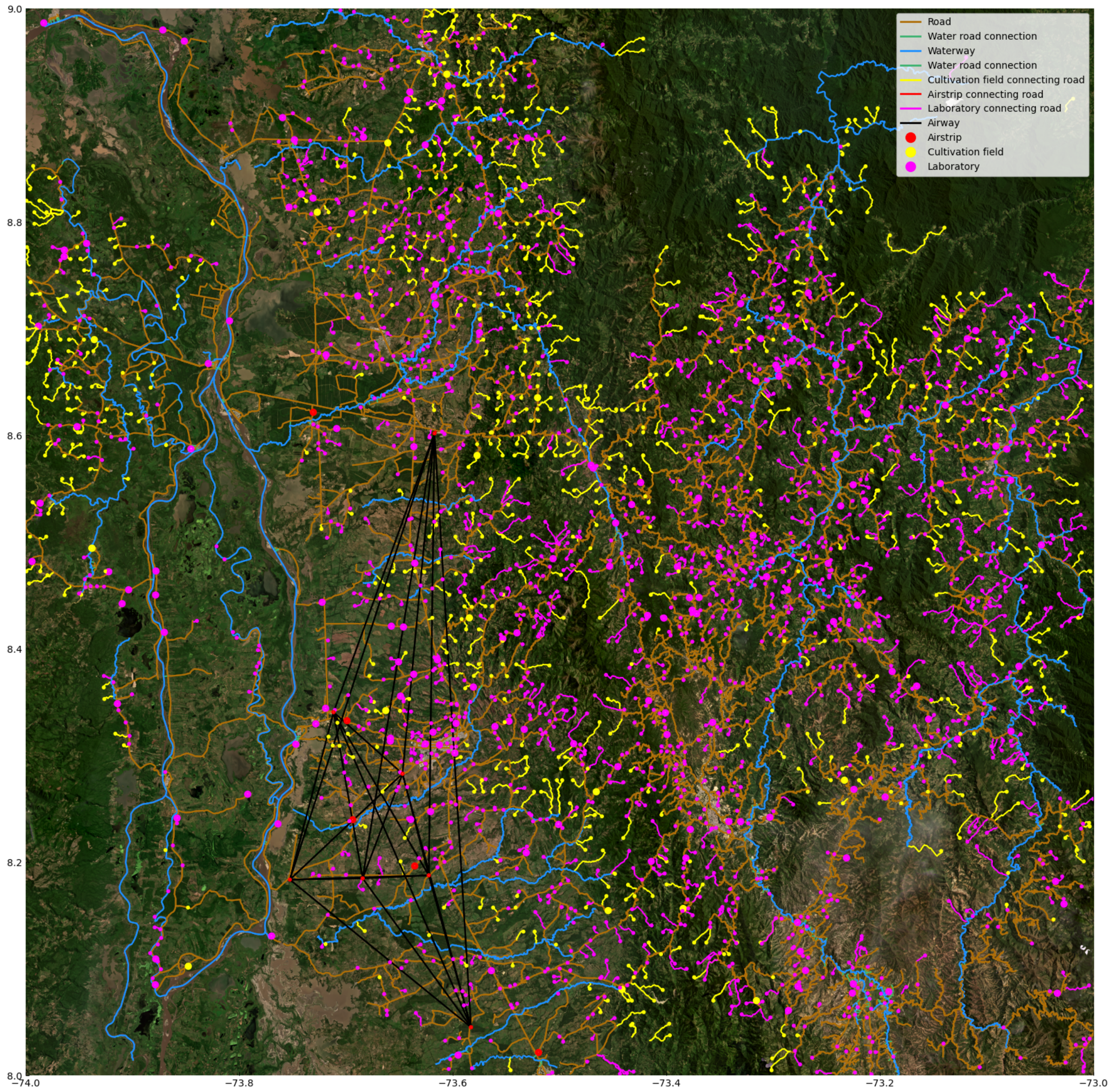


Figure 6.16: Graph with all nodes added through pathfinding. Image adapted from Planet Labs PBC (2024b).

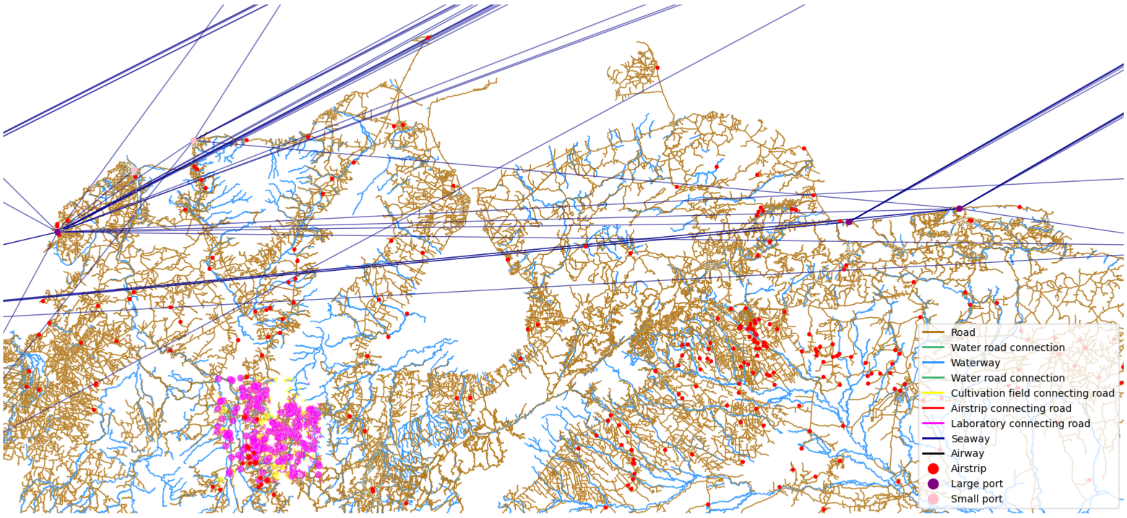


Figure 6.17: Graph with connections between roads, waterways, ports, airstrips, cultivation fields, and laboratories

6.4 Identifying routes

A fully connected graph is required to calculate routes. Consequently, the largest fully connected component of the graph is selected, resulting in the loss of approximately 300 laboratories and 75 fields. After this selection, 2083 laboratories and 855 fields remain. In this phase, edge weights are computed, requiring the assignment of transport modalities (truck and donkey). A large elevation raster is constructed by connecting smaller elevation tiles. The result is depicted in Figure 6.18. Using the OSMNX tool, elevation values are assigned to graph nodes, which are then converted to slope values for the edges. Then, the edge weights are calculated according to the formulas outlined in Chapter 5.

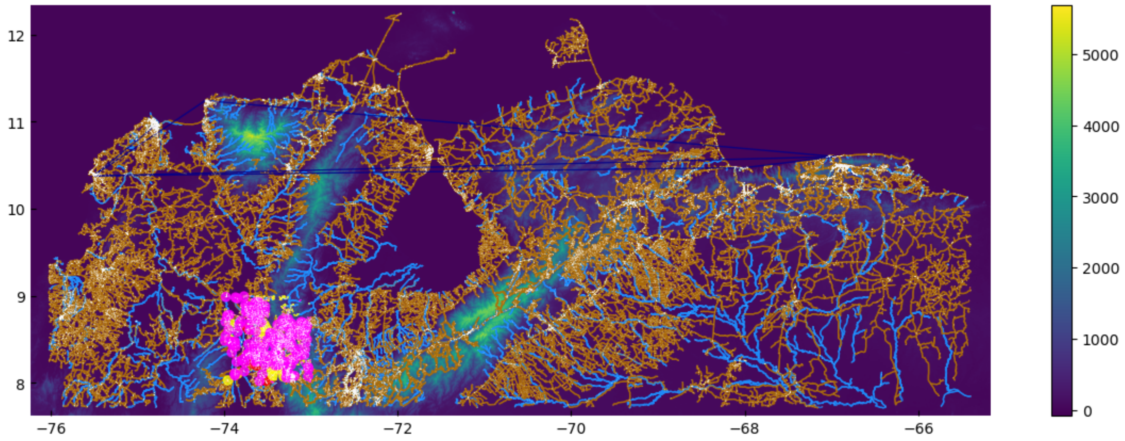


Figure 6.18: Graph on large elevation raster. Data adapted from NASA (2024).

6.4.1 From fields to laboratories

Path calculations are restricted to field-laboratory pairs within a 10-kilometre radius to reduce computation time. This limit is realistic, as small laboratories are typically located near fields to facilitate processing closer to the source, given that transporting processed leaves is more efficient than raw crops. The restriction is unlikely to significantly affect model outcomes, as each field has multiple laboratories within this range. Additionally, the shortest path algorithm is unlikely to identify shorter paths to laboratories outside this radius, particularly for weights such as travel time and cost. While the restriction may slightly affect outcomes for the weight ”risk,” where distant laboratories could be accessed through fewer edges, this impact is expected to be minimal. The 10-kilometre limit also aligns with real-world scenarios where plane usage is improbable at this stage of the supply chain.

After applying this range limit, there are approximately 55,000 origin-destination pairs between which the shortest paths are computed. After calculating the shortest paths for each pair, a single laboratory is selected for each field based on the shortest path distance. The resulting paths, calculated for each weight, are presented in Figure 6.19.

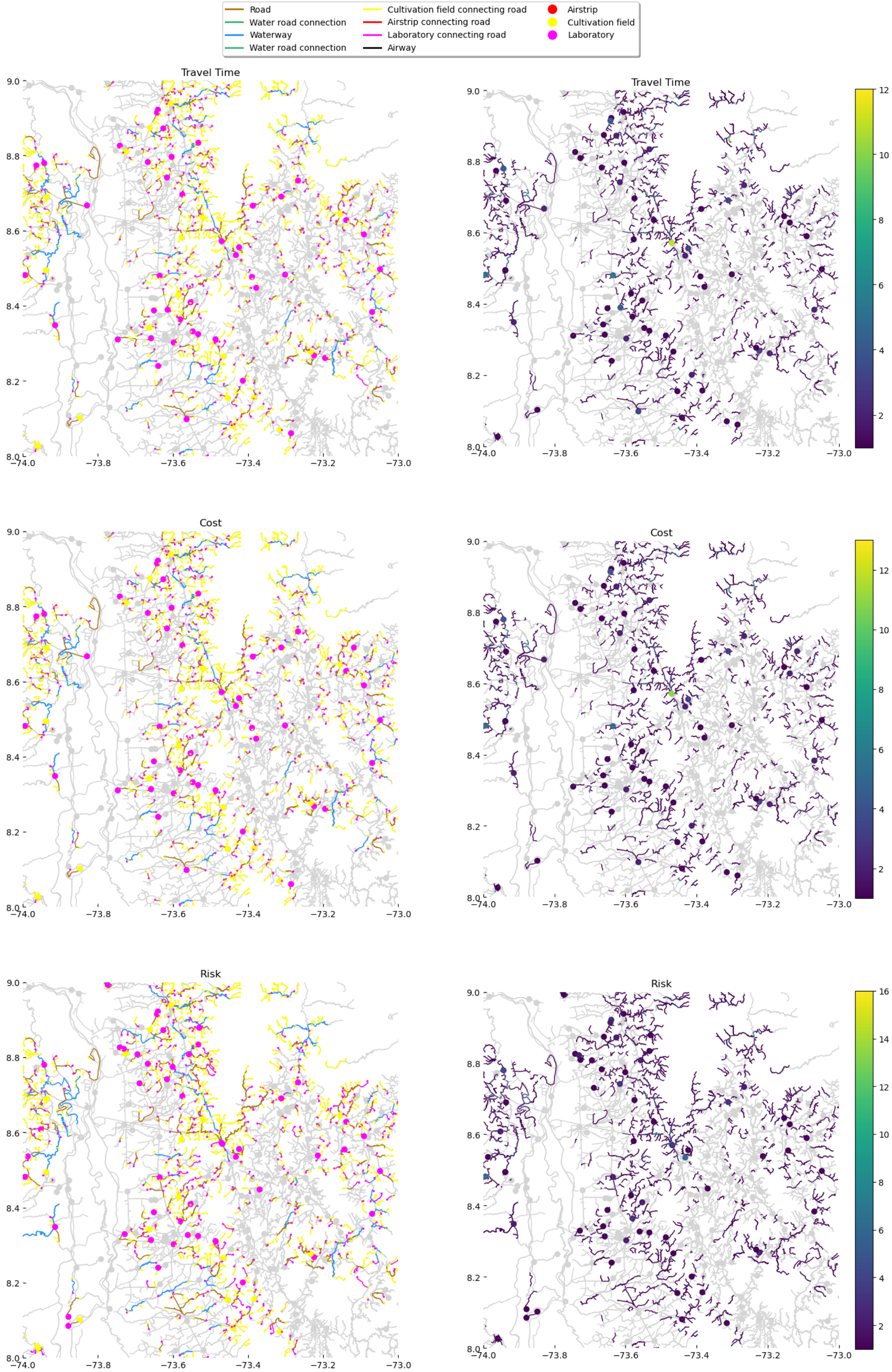


Figure 6.19: Graph with routes and edge frequencies from fields to laboratories.

Minimal visual differences are observed between the three weights; however, Figure 6.20 (and its detailed counterpart in Appendix B, Figure B.1) highlights a subtle distinction, indicating that the risk metric deviates most significantly from cost and travel time. Notably, in the middle section of the graph, the route optimized for risk leads to a different laboratory than the routes optimized for cost or travel time. Although the underlying reason for this divergence is unclear from this graph, further insights are provided in Figure 6.21.

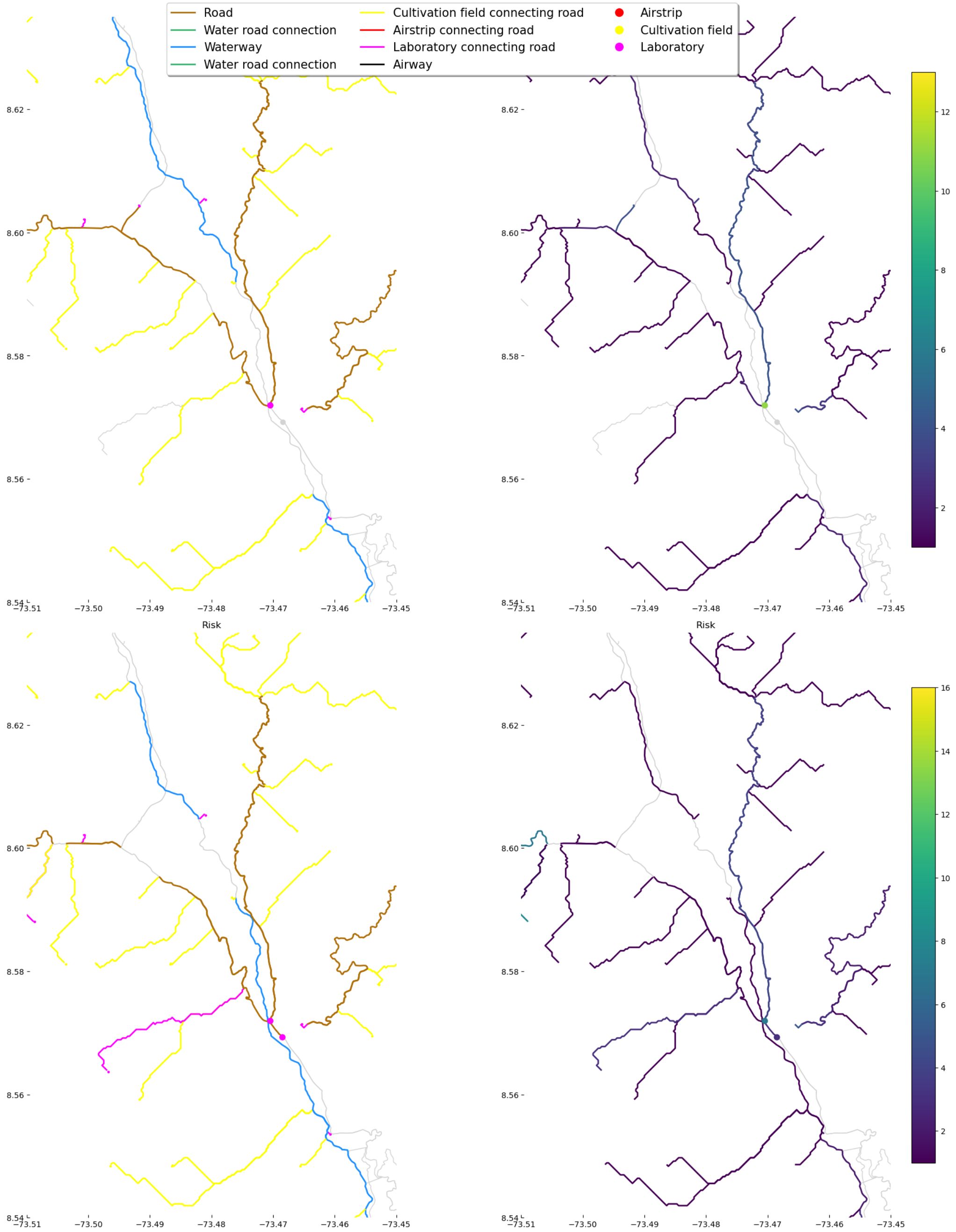


Figure 6.20: Comparison routes and edge frequencies from fields to laboratories between cost and risk.

Figure 6.21 shows the frequency of modality usage, revealing a lower count for the risk metric compared to the cost and travel time metrics, which exhibit relatively similar frequencies. Notably, edges with zero length are classified under the "no modality" category, and the total edge count corresponds to the combined modality count. These findings indicate that risk-optimized paths favour routes with fewer edges, while cost and travel time prioritize shorter path distances.

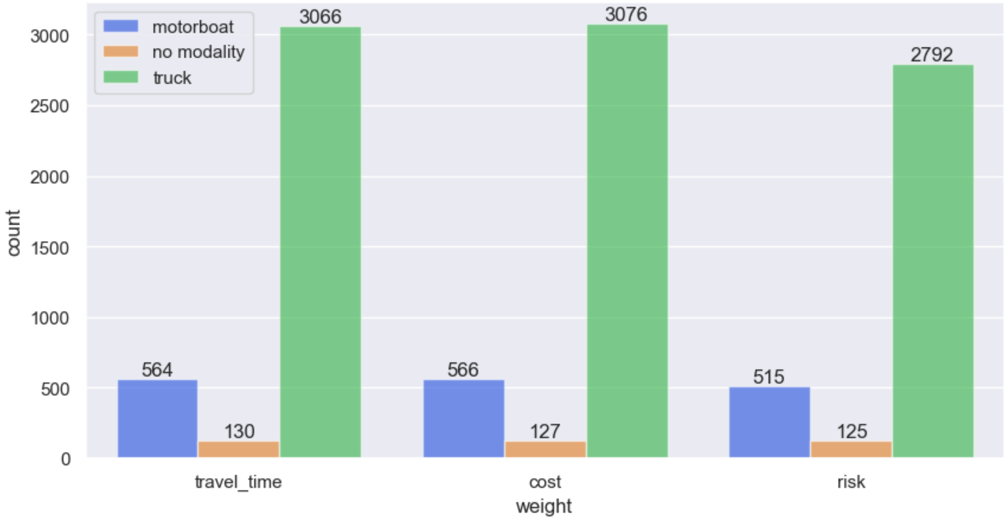


Figure 6.21: Frequency of modality usage by weight type: routes from cultivation fields to laboratories.

Additionally, Figure 6.22 compares the total cost, travel time, and risk for each weight type. The data reinforce that routes optimized for cost and travel time perform similarly across all metrics, whereas those optimized for risk result in higher cumulative costs and travel times. This observation underscores the trade-off between cost and risk, as discussed by Klaassen (2021). In contrast, there is minimal trade-off between cost and travel time for these short-distance paths.

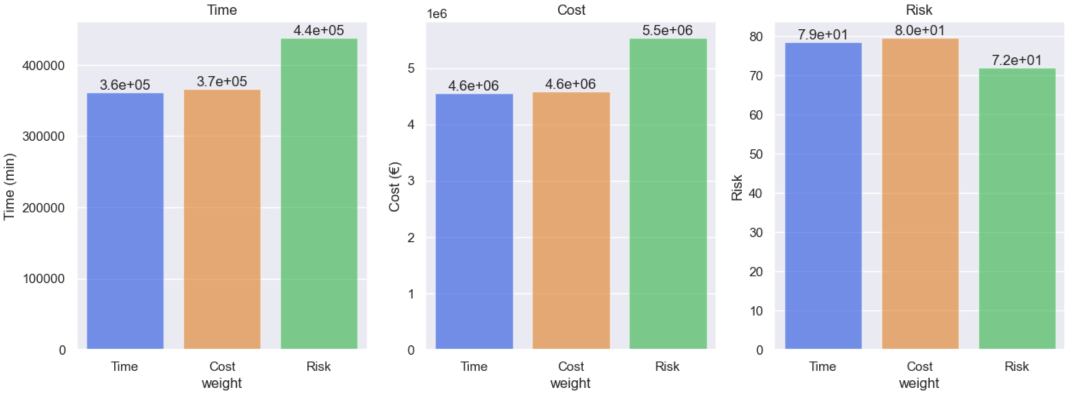


Figure 6.22: Comparison of travel time, cost, and risk by weight type: routes from cultivation fields to laboratories.

6.4.2 From laboratories to ports

Only laboratories with the shortest paths from fields are included in this analysis, totalling 616 laboratories. Six European ports, varying in size, are selected as destinations: Rotterdam (NLRTM), Vlissingen (NLVLI), Antwerp (BEANR), Zeebrugge (BEZEE), Hamburg (DEHAM), Algeciras (ESALG) and Le Havre (FRLEH). Origin-destination pairs are established between these ports and the selected laboratories.

For each laboratory, the path to the port with the shortest path among all options is selected. The results are presented in Figure 6.23.

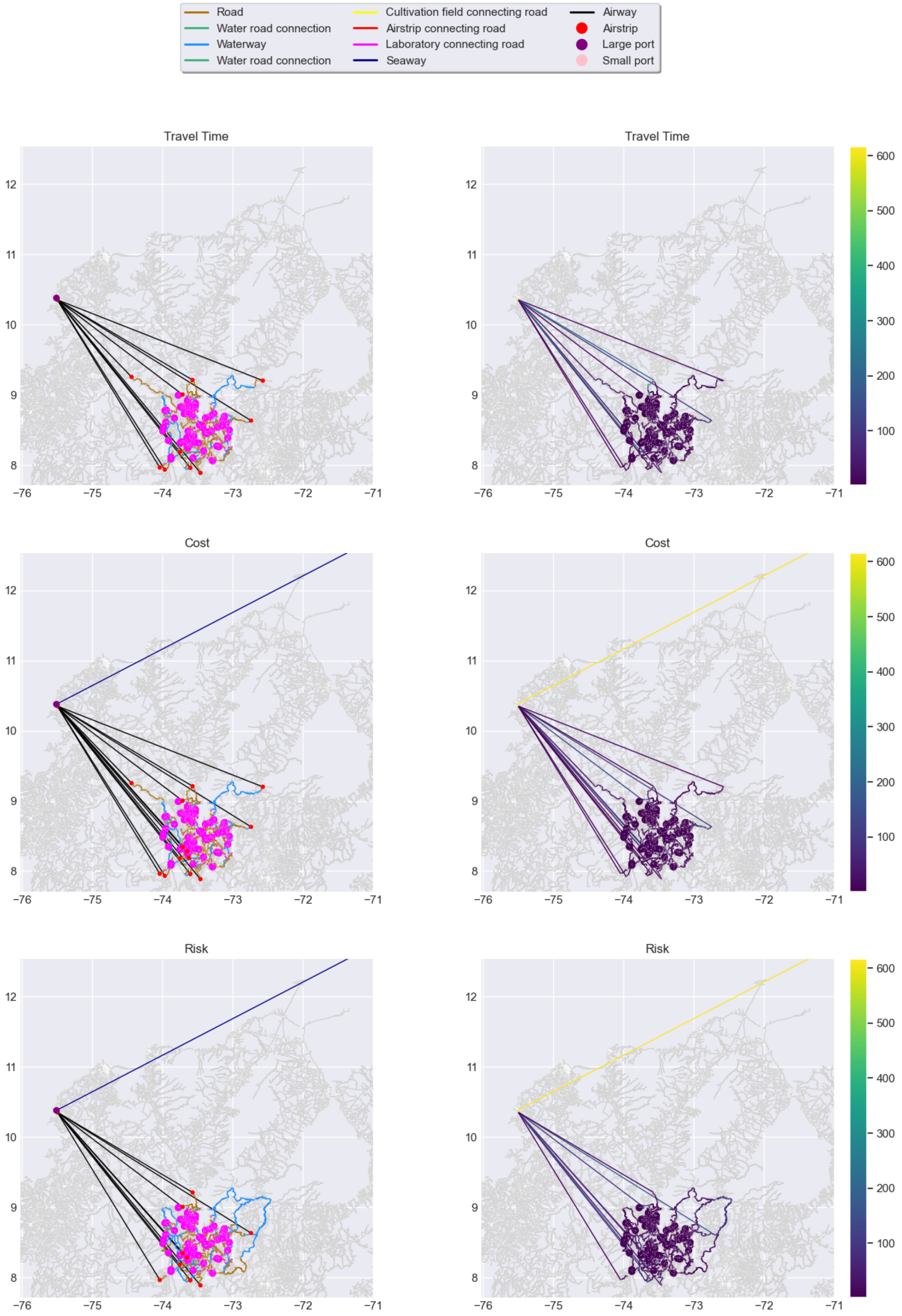


Figure 6.23: Graph with routes and edge frequencies from laboratories to ports.

Figure 6.23 presents graphs depicting the shortest routes and corresponding edge frequencies between laboratories and ports, optimized for three metrics: cost, travel time, and risk. Across all metrics, the routes consistently go through the same port, Cartagena in Colombia. This could indicate its centrality in the supply chain in the case study area, but it could also be a limitation of the chosen route selection model, as this port is also the nearest port to the case study area.

For the travel time metric, the arrival port in Europe for all routes is Algeciras, Spain. In contrast, the cost and risk metrics do not have a clear preference for a specific port, except for Vlissingen, which is excluded due to the absence of a direct connection to Cartagena. This outcome is a result of the equations employed for these metrics, where distance does not influence cost or risk on sea edges. Moreover, no distinction is made between larger and smaller ports in Europe, further contributing to the lack of specificity in port selection under these metrics.

The routing patterns for cost and travel time are visually similar. However, differences in edge frequency are evident, as highlighted in Figure B.2 in Appendix B. Figure 6.24, which depicts the count of each transport modality by weight type, further supports these differences. For the travel time metric, trucks are utilized more frequently than motorboats, in contrast to the cost metric. This observation is somewhat unexpected, as motorboats have higher costs per kilometre than trucks. A plausible explanation is that the average speed on selected routes for travel time optimization is higher, favouring the use of trucks for the travel time metric.

Routes optimized for risk show notable deviations from those for cost and travel time. Figure 6.24 also shows that the risk metric favours fewer edges for longer routes and exhibits a stronger preference for motorboats compared to the other metrics. This preference may arise from the ability to traverse waterways for extended distances without the need for crossings or the additional risk associated with transloading onto trucks, thereby minimizing disruptions and the risk of getting noticed.

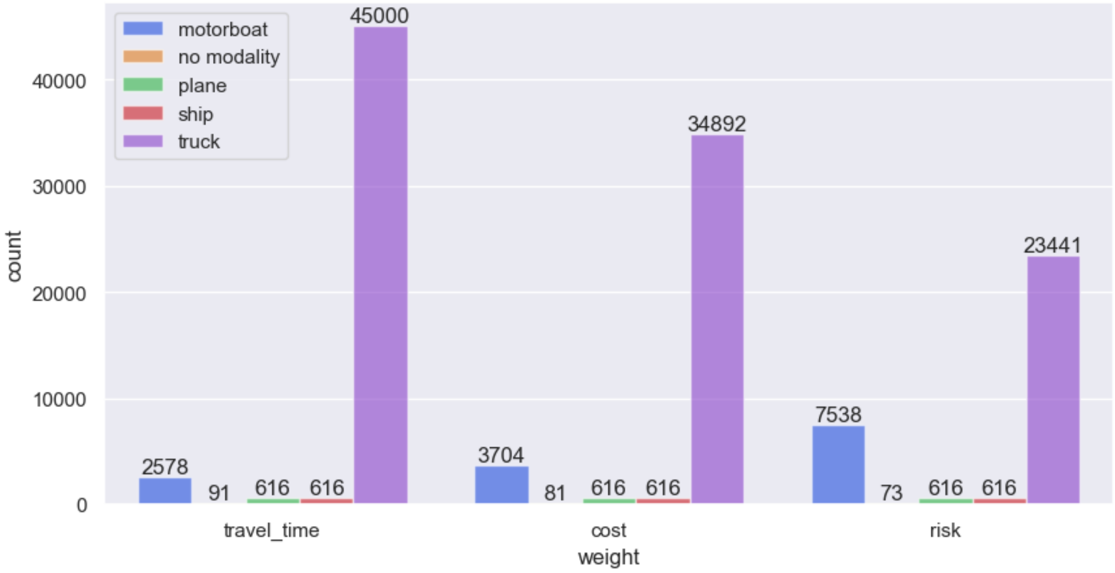


Figure 6.24: Frequency of modality usage by weight type for routes from laboratories to ports.

All metrics consistently utilize aircraft and ships 616 times, indicating that every path includes an aircraft segment to reach a port and a single maritime segment to connect to a European port. This suggests that the higher risk or cost associated with air travel is offset by the greater risk or cost of traversing such long distances via ground transport.

Figure 6.25 summarizes the total cost, travel time, and risk for each type of shortest path weight. The travel time and cost metrics remain closely aligned across all weights, while the risk metric diverges most prominently. This observation underscores the existence of a trade-off between risk and cost, even for longer routes, in the selection of optimal paths.

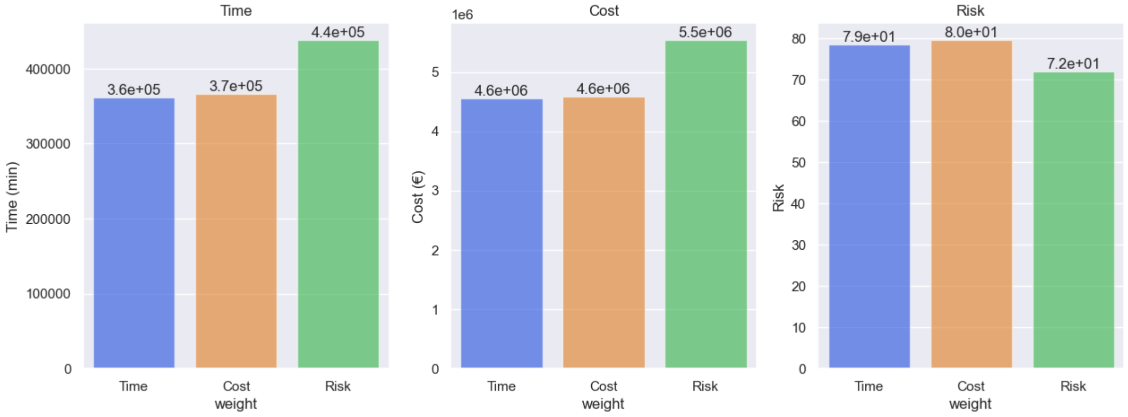


Figure 6.25: Comparison of travel time, cost, and risk by weight type: routes from laboratories to ports.

7 | Discussion

This study developed a proof of concept for using geospatial intelligence (GEOINT) and machine learning to identify and analyse routes in illicit supply chains. The study offers a framework for monitoring these illicit networks by leveraging satellite data, constructing a graph and estimating routes. The following discussion synthesizes insights, limitations, and potential directions for future research in each area of the study.

7.1 The approach

A conceptual framework was developed to model illicit supply chains using graph theory, which allowed for the analysis of the inherently spatial nature of these activities. This framework identified key nodes such as cultivation fields, laboratories, and ports, with edges representing potential smuggling routes based on research by Aschner & Montero (2021) on the infrastructure in illicit supply chains. By integrating machine learning with graph theory and real-world supply chain routing, the framework introduced a novel approach that provides insights not only into the production locations of goods but also their movement through the network. This methodology is particularly valuable for gaining a deeper understanding of illicit supply chains, even when complete data is unavailable. Furthermore, the framework's design is modular and adaptable, allowing for easy updates or modifications to accommodate evolving research needs.

The framework also has some notable limitations. The approach does not account for the complex, adaptive behaviours of illicit networks, such as their ability to respond dynamically to enforcement efforts, which are critical for reflecting real-world applicability. Key components, including distribution centres and large-scale laboratories, are not currently included, which reduces the model's comprehensiveness.

Future research should aim to address these limitations by incorporating dynamic elements, such as the influence of enforcement activities and spatial factors on route selection, to better capture the complexities of illicit networks. Expanding the graph to include larger laboratories, inland ports, and distribution centres would also enhance its scope.

7.1.1 Machine learning models

Machine learning models demonstrated the ability to detect potential cultivation fields and laboratories from satellite imagery, highlighting the promise of GEOINT for identifying objects related to potential illicit activities in remote areas. Although there was not a model designed for detecting potentially illicit cultivation fields publicly available, this research introduced an approach to adapt the model by Maladière (2022) for such purposes. While GEOINT has been applied in this context before (Pinto Hidalgo & Silva Centeno, 2022), the study reaffirmed its utility and provided valuable insights.

Despite these achievements, the analysis faced notable challenges. The training datasets were limited in their ability to distinguish between licit and illicit cultivation fields, which can lead to false positives. Additionally, the model overestimated the number of suspected illicit laboratories, likely due to structural similarities between illicit laboratories and other types of buildings. These limitations underscore the need for refinement.

Future research should focus on improving datasets to more effectively differentiate licit from illicit land uses, incorporating a wider variety of geographic and environmental conditions to enhance model generalization. Stricter or additional selection criteria could further refine detection accuracy, particularly in distinguishing illicit laboratories from similar structures. These improvements would bolster the reliability of GEOINT-based approaches for monitoring objects potentially related to illicit activities.

7.1.2 Graph construction

The study constructed a graph combining licit and illicit infrastructure, including roads, waterways, and airstrips, to estimate smuggling routes. This graph incorporated both known and inferred infrastructure, providing a comprehensive representation of possible transport pathways. A notable innovation was the use of slope data to approximate undocumented roads connecting remote cultivation fields and laboratories, significantly enhancing the model's realism.

However, the process faced challenges, including gaps in OpenStreetMap (OSM) data for illicit airstrips and unofficial roads and the computational demands of approximating undocumented roads based on slope data, which limited the ability to identify paths in the most remote areas. Additionally, all waterways were assumed to be navigable, which may not reflect reality due to obstacles such as insufficient depth or waterfalls.

Future improvements could address these limitations by using alternative or locally-sourced datasets to fill gaps in infrastructure data and enhance graph accuracy. Methods to assess waterway navigability and selectively include travelable edges, such as lakes and waterways passable by motorboats, could further improve graph validity. Moreover, refining the sequence for connecting nodes and edges could address current oversights where nearby fields or laboratories remain unconnected due to their proximity to airstrips rather than roads or waterways. This would ensure a more realistic representation of the network.

Finally, integrating advanced machine learning techniques could help detect and incorporate unofficial roads, illicit airstrips, and unauthorized flights, improving the graph's precision and expanding its utility for analysing smuggling networks.

7.2 Route identification

The study employed Weighted Dijkstra's algorithm to estimate optimal smuggling routes based on travel time, cost, and risk. All though, Dijkstra's has been applied before estimating routes in illicit supply chains by Achi et al. (2012), they only applied it with a travel time weight. Equations were developed to calculate the travel time, cost, and risk values for each edge, enabling a direct comparison of these metrics.

However, there are some limitations to this route identification approach. The study's risk calculations were simplified, omitting critical factors such as political control, terrain characteristics, and enforcement presence in risk levels. Additionally, the deterministic nature of Dijkstra's algorithm limited its ability to model chance occurrences, the dispersion of risk across multiple routes, or adaptive behaviours in dynamic scenarios. The current model does not incorporate differentiated weights for arrival ports in Western countries, limiting its capacity to provide valuable insights into the selection and significance of specific arrival ports.

Future research could address these limitations by adopting graph-based agent-based modelling (ABM), which allows for more complex and adaptive dynamics within the graph. Incorporating more sophisticated risk assessments or scenario modelling, would enhance the accuracy and applicability of risk calculations. The inclusion of additional transport modalities could further improve the model, such as differentiating between larger vessels and feeder ships in maritime routes, different types of trucks or adding semi-submersibles. These would enable to calculate more precise edge travel times, cost and risks. It is also recommended to incorporate differentiated weights for arrival ports in Western countries. These weights should reflect factors such as enforcement intensity, proximity to major markets and efficiency.

7.3 Case study

The application of graph and route estimation methods to a case study of the Colombia-Venezuela border demonstrated the model's capacity to identify plausible smuggling routes and potential illicit nodes within a real-world context, providing a valuable proof of concept for future research. Analysis revealed significant divergence between risk-weighted routes and those optimized for cost and time, underscoring the importance of incorporating risk metrics into the analysis of route selection within illicit networks.

This observed trade-off between cost- and time-weighted routes and risk is consistent with findings from previous research (Klaassen, 2021). Although the current risk metric accounts only for modality-based risk and the risk of transloading goods, it sufficiently captures the trade-offs that are reflective of real-world smuggling routes in South America. Enhancements to the risk metric, such as integrating factors like political control, terrain characteristics, and law enforcement presence—are likely to amplify the differences in routes, as these factors introduce location-specific variations not accounted for in time and cost calculations, which are primarily influenced by edge types and modalities.

The existence of a trade-off between cost- and time-weighted routes is less evident in the results. While there are minor variations in edge frequencies, the overall travel times, costs and risks of the routes remain similar. Incorporating additional parameters, such as the acquisition costs of transportation modalities or refined travel times for maritime routes based on actual shipment data, rather than distance alone, could reveal greater distinctions. Further investigation is required to explore these dynamics in more detail.

An unexpected finding was that all routes, irrespective of weighting metrics, converged on the same port in Colombia, with no observed transit through multiple ports or Venezuela. This outcome does not align with real-world practices, where smuggling networks typically distribute goods across multiple ports and routes, including transits through Venezuela.

Several factors may explain this discrepancy. First, the exclusion of smaller ports, such as the port of Maracaibo, which is located near the case study area, may have limited the scope of the model. Including these ports in future analyses could yield more realistic results and reflect transnational transportation dynamics. Second, the estimation of weights, particularly for risk, may require further refinement. Real-world smuggling routes often favour ports with less law-enforcement presence and less technological advancement, a factor not fully captured in the current model. Third, the deterministic nature of the shortest path algorithm does not account for the dispersion of goods across multiple routes and ports, therefore not including all ports through which goods are transported in reality.

To address these issues, future research should conduct comparative analyses of routes to ports in Europe, Colombia, and Venezuela to provide deeper insights into route selection and international smuggling dynamics. Sensitivity analyses of weight parameters, particularly risk, could further clarify the extent to which adjustments influence route outcomes and identify thresholds for significant changes.

A notable limitation of the case study was the lack of robust validation. Relying solely on satellite imagery was insufficient for comprehensive verification of the findings. Future research should enhance validation efforts by incorporating ground-level data, local law enforcement records, expert knowledge, and secondary data sources. This additional data or knowledge would not only strengthen the model's accuracy but also support more reliable estimation of weight values.

The model also has the potential to capture temporal dynamics by analysing snapshots of the network at different points in time using satellite datasets from various years. This capability could facilitate comparisons over time, enabling the detection of changes in network structures and informing adaptive intervention strategies based on evolving smuggling patterns.

7.4 General conclusion

In conclusion, this study contributes a novel application of GEOINT to understanding illicit supply chains. While the initial findings are promising, future research and model refinements are necessary to improve detection accuracy and better account for real-world complexities.

8 | Conclusion

This research explores how satellite data combined with machine learning models can be applied to identify and analyse routes within illicit supply chains, particularly focusing on the South America-Europe trade corridor. The primary research question is:

How can satellite data combined with machine learning models be used to identify plausible routes based on illicit and licit infrastructure networks?

The study addresses this question by answering the following sub-questions.

1. To what extent are machine learning models suitable for identifying plausible illicit nodes within the infrastructure graph, given open-source satellite imagery and georeferenced data?

The study found that machine learning models are a promising tool for identifying potential illicit nodes, such as cultivation fields and laboratories, from satellite imagery. The models demonstrated capability in detecting these nodes within large, remote areas, thus confirming the potential of GEOINT to identify objects potentially related to illicit activities. However, limitations emerged regarding dataset specificity; distinguishing between licit and illicit nodes remains challenging due to similar visual features and limited labelled training data specific to illicit uses. Due to the inherent uncertainties in the available datasets, it is only possible to identify potentially illicit object, and not conclusively identify the objects.

2. How can illicit nodes and edges be combined with licit infrastructure into a graph?

A graph was constructed, integrating both licit infrastructure (such as roads and waterways) and detected illicit nodes. By connecting remote fields and laboratories with licit roads, airstrips, and waterways, the graph enables the exploration of plausible routes from cultivation sites to ports. Notably, including slope data to infer undocumented paths between remote nodes added realism to the model. However, data limitations from sources like OpenStreetMap (OSM) led to gaps, and integrating locally sourced data, identifying more objects with machine learning or alternative datasets could further improve graph accuracy and completeness.

3. How can plausible routes be identified given the identified combined graph of the licit and illicit infrastructure?

The study applied a weighted Dijkstra's algorithm to the combined graph to estimate routes based on travel time, cost, and risk. This approach effectively identified plausible routes and highlighted differences in route selection when optimized for the different criteria. However, Dijkstra's deterministic nature limited the model's ability to accommodate variability, such as chance-based risk dispersion or dynamic scenarios. Future research could address these limitations with a graph-based agent-based model, which is more suitable for modelling dynamic route choices.

4. Given this approach, what are the plausible routes based on the combined

licit and illicit infrastructure network in the Colombia-Venezuela border region?

The application of graph-based route estimation methods to the Colombia-Venezuela border region identified plausible smuggling routes connecting cultivation fields to international ports. The analysis indicated that all routes converge at the Port of Cartagena, Colombia, with air transportation consistently utilized to reach this destination. When examining the subsequent transport to international ports, the travel time metric exhibited a clear preference for the arrival port at Algeciras, Spain, while the cost and risk metrics did not demonstrate a similarly defined preference, due to the structure of their respective metric equations.

Notably, routes optimized for risk diverged significantly from those optimized for cost or time. This divergence was observed both in the pathways connecting cultivation fields to laboratories and in those linking laboratories to ports. These findings suggest a trade-off between cost and risk in route selection, highlighting the importance of incorporating risk metrics into models of illicit network dynamics.

While these results provide valuable insights, additional validation is required to assess their real-world applicability. Incorporating field data, law enforcement records, and other secondary sources would enhance the robustness of the findings and ensure their relevance to actual smuggling networks.

Overall Contribution and Future Directions

In summary, this research demonstrates a viable proof of concept for applying GEOINT and graph-based analysis to model and analyse illicit supply chains. The findings underscore the utility of open-source satellite data and machine learning in tracking and monitoring these clandestine networks, and provide a foundation for future developments in illicit network modelling. Future research should focus on refining detection algorithms, enhancing risk models, and integrating multi-source intelligence to improve the accuracy and practical applicability of this approach in real-world scenarios. By continuing to develop these tools, this research contributes to the growing field of data-driven, scalable intelligence solutions in countering illicit trade networks.

References

Achi, H., Adeofun, C., Ufoegbune, G., Gbadebo, A., & Oyedepo, J. (2012). Disposal sites and transport route selection using geographic information system and remote sensing in Abeokuta, Nigeria. Global Journal of Human Social Science, 12(12), 14–23.

Agarwal, M. (2024). What is The Speed of a Ship at Sea? https://www.marineinsight.com/guidelines/speed-of-a-ship-at-sea/.

Alake, R. (2023). Loss functions in machine learning explained. Retrieved 2024-10-01, from https://www.datacamp.com/tutorial/loss-function-in-machine-learning

Anderson, C., & Potter, H. (2022). Amazônia tem 362 pistas de pouso clandestinas perto de áreas devastadas pelo garimpo. Retrieved from https://www.intercept.com.br/2022/08/02/amazonia-pistas-clandestinas-garimpo/

Ansel, J., Yang, E., He, H., Gimelshein, N., Jain, A., Voznesensky, M., … Chintala, S. (2024). PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation. In 29th acm international conference on architectural support for programming languages and operating systems, volume 2 (asplos '24). ACM. Retrieved from https://pytorch.org/assets/pytorch2-2.pdf doi: 10.1145/3620665.3640366

Aschner, J. P., & Montero, J. C. (2021). Architectures, spaces, and territories of illicit drug trafficking in Colombia and Mexico. Crime, Media, Culture, 17(3), 327–351. doi: doi.org/10.1177/1741659020910212

Becerra, J., Ariza, A., & Gamarra-Amaya, L. C. (2021). Use of open-source satellite data to combat organized crime case study: Detection of vessels associated with drug-trafficking. In A. Froehlich (Ed.), Space fostering latin american societies: Developing the Latin American continent through space, part 2 (pp. 67–86). Cham: Springer International Publishing. doi: doi.org/10.1007/978-3-030-73287-5_5

Berkson, E., Groener, A., Cuellar-Vite, C., Chern, G., O'Neill, S., Harner, M., … Pritt, M. (2020). Methods of exploiting multispectral imagery for the monitoring of illicit coca fields. In 2020 ieee applied imagery pattern recognition workshop (aipr) (p. 1-11). doi: 10.1109/AIPR50011.2020.9425056

BoatDriving.org. (2024). How Fast Can a Boat Go? (Chart). https://www.boatdriving.org/how-fast-do-speed-boats-go/.

Boeing, G. (2024). Modeling and analyzing urban networks and amenities with osmnx. Retrieved from https://geoffboeing.com/publications/osmnx-paper/

Bolívar-Santamaría, S., & Reu, B. (2021). Detection and characterization of agroforestry systems in the Colombian Andes using Sentinel-2 imagery. Agroforestry Systems, 95(3), 499–514. doi: doi.org/10.1007/s10457-021-00597-8

Botelho, J., Costa, S. C. P., Ribeiro, J. G., & Souza, C. M. (2022). Mapping roads in the Brazilian Amazon with artificial intelligence and Sentinel-2. Remote Sensing, 14(15), Article 3625. doi: doi.org/10.3390/rs14153625

Brownlee, J. (2020). A gentle introduction to the fbeta-measure for machine learning. Retrieved 2024-10-01, from https://machinelearningmastery.com/fbeta-measure-for-machine-learning/

Chadid, M., Davalos, L., Molina Escobar, J., & Armenteras, D. (2015). A bayesian spatial model highlights distinct dynamics in deforestation from coca and pastures in an andean biodiversity hotspot. Forests, 6, 3828-3846. doi: doi.org/10.3390/f6113828

DEA. (1991). Coca cultivation and cocaine processing: An overview (No. 132907). NCJR. Retrieved from https://www.ojp.gov/ncjrs/virtual-library/abstracts/coca-cultivation-and-cocaine-processing-overview

Emily, J. A., & Sudha, N. (2022). Case studies: Deep learning in remote sensing. In Fundamentals and methods of machine and deep learning (p. 425-437). John Wiley & Sons, Ltd. doi: doi.org/10.1002/9781119821908.ch18

Escobar-López, A., Castillo-Santiago, M. Á., Hernández-Stefanoni, J. L., Mas, J. F., & López-Martínez, J. O. (2022). Identifying coffee agroforestry system types using multitemporal Sentinel-2 data and auxiliary information. Remote Sensing, 14(16), Article 3847. doi: doi.org/10.3390/rs14163847

Est, L. (2022). Coca cultivation in peru: A geospatial intelligence collection plan. Retrieved from http://jhir.library.jhu.edu/handle/1774.2/67818

Gallimore, D. (2024). The national average salary in Colombia. https://www.outsourceaccelerator.com/articles/average-salary-in-colombia/.

Goldenberg, B., Uzkent, B., Clough, C., Funke, D., Desai, D., Grischa, ... Kan, W. (2017). Planet: Understanding the amazon from space. Kaggle. Retrieved from https://kaggle.com/competitions/planet-understanding-the-amazon-from-space

González-González, A., Clerici, N., & Quesada, B. (2022). A 30 m-resolution land use-land cover product for the colombian andes and amazon using cloud-computing. International Journal of Applied Earth Observation and Geoinformation, 107, 102688. doi: doi.org/10.1016/j.jag.2022.102688

Hagberg, A. A., Schult, D. A., & Swart, P. J. (2008). Exploring network structure, dynamics, and function using networkx. In G. Varoquaux, T. Vaught, & J. Millman (Eds.), Proceedings of the 7th python in science conference (p. 11 - 15). Pasadena, CA USA.

Klaassen, R. (2021). The route of crime. Retrieved from https://resolver.tudelft.nl/uuid:c1bce36d-1c92-4657-b410-83b29336fac6

LightningAI. (2024). Mean-average-precision (map). Retrieved 2024-10-01, from https://lightning.ai/docs/torchmetrics/stable/detection/mean_average_precision.html

Magliocca, N. R., McSweeney, K., Sesnie, S. E., Tellman, E., Devine, J. A., Nielsen, E. A., ... Wrathall, D. J. (2019). Modeling cocaine traffickers and counterdrug interdiction forces as a complex adaptive system. Proceedings of the National Academy of Sciences, 116(16), 7784–7792. doi: doi.org/10.1073/pnas.1812459116

Magliocca, N. R., Summers, D. S., Curtin, K. M., McSweeney, K., & Price, A. N. (2022). Shifting landscape suitability for cocaine trafficking through Central America in response to counterdrug interdiction. Landscape and Urban Planning, 221, Article 104359. doi: doi.org/10.1016/j.landurbplan.2022.104359

makepath. (2024). Xarray spatial. Retrieved from https://github.com/makepath/xarray-spatial

Maladière, V. (2022). Pytorch resnet18 [0.93]. Retrieved from https://www.kaggle.com/code/vincentmaladiere/pytorch-resnet18-0-93

Montagut Llauradó, M. (2022). Hiking networks generation from elevation maps (Unpublished master's thesis). Universitat Politècnica de Catalunya.

NASA. (2024). Earthdata search. Retrieved 2024-10-04, from https://search.earthdata.nasa.gov

Nazarova, T., Martin, P., & Giuliani, G. (2020). Monitoring vegetation change in the presence of high cloud cover with Sentinel-2 in a lowland tropical forest region in Brazil. Remote Sensing, 12(11), Article 1829. doi: doi.org/10.3390/rs12111829

NGA. (2018). Geospatial intelligence (geoint) basic doctrine. National Geospatial-Intelligence Agency. Retrieved from https://www.nga.mil/resources/GEOINT_Basic_Doctrine_Publication_10_.html

Niamkaeo, S., & Robert, O. (2020). Spatial relationship of drug smuggling in northern Thailand using GIS-based knowledge discovery. Environment and Natural Resources Journal, 18(3), 275–282. doi: doi.org/10.32526/ennrj.18.3.2020.26

OpenStreetMap contributors. (2017). Planet dump retrieved from https://planet.osm.org. https://www.openstreetmap.org.

OSCE. (2017). Osce guidebook intelligence-led policing. Organization for Security and Co-operation in Europe. Retrieved from https://www.osce.org/chairmanship/327476

Pacheco-Pascagaza, A., Gou, Y., Louis, V., Roberts, J., Rodríguez-Veiga, P., da Conceição Bispo, P., ... Balzter, H. (2022). Near real-time change detection system using Sentinel-2 and machine learning: a test for Mexican and Colombian forests. Remote Sensing, 14(3), 275-–282. doi: doi.org/10.3390/rs14030707

Pinto, J. (2022). Cocapaste-pi-detection dataset. Mendeley Data. doi: 10.17632/gmhsjwr24n.1

Pinto Hidalgo, J. J., & Silva Centeno, J. A. (2022). Geospatial intelligence and artificial intelligence for detecting potential coca paste production infrastructure in the border region of Venezuela and Colombia. Journal of Applied Security Research, 1–51. doi: doi.org/10.1080/19361610.2022.2111184

Planet Labs PBC. (2024a). Norway's international climate and forests initiative satellite data program. Retrieved from https://www.planet.com/nicfi/

Planet Labs PBC. (2024b). Planet application program interface: In space for life on earth. Retrieved from https://api.planet.com

Prato, C. G. (2009). Route choice modeling: past, present and future research directions. Journal of Choice Modelling, 2(1), 65-100. doi: doi.org/10.1016/S1755-5345(13)70005-8

Prices, G. P. (2024). Colombia precios de la gasolina, 21-oct-2024 | GlobalPetrolPrices.com. https://es.globalpetrolprices.com/Colombia/gasoline_prices/.

Rath, S. R. (2023). Faster r-cnn pytorch training pipeline. GitHub. Retrieved from https://github.com/sovit-123/fasterrcnn-pytorch-training-pipeline

Robert, O. P., Witheetrirong, Y., Janpengpen, A., & Kittikachorn, C. C. (2015). Defense geo-database: Drug trafficking. In Proceedings of the 36th asian conference on remote sensing (pp. 659–664). doi: dx.doi.org/10.13140/RG.2.1.3451.8484

Scikit-learn. (2024). sklearn.metrics.fbeta_score. Retrieved 2024-10-01, from https://docs.w3cub.com/scikit_learn/modules/generated/sklearn.metrics.fbeta_score.html

Shafran, D. (2024). How Much Fuel Does A Boat Use Per Hour? Guide + Examples. https://maritimepage.com/how-much-fuel-does-a-boat-use/.

Shah, D. (2022). Mean average precision (map) explained: Everything you need to know. Retrieved 2024-10-01, from https://www.v7labs.com/blog/mean-average-precision

Shendryk, Y., Rist, Y., Ticehurst, C., & Thorburn, P. (2019). Deep learning for multimodal classification of cloud, shadow and land cover scenes in planetscope and sentinel-2 imagery. ISPRS Journal of Photogrammetry and Remote Sensing, 157, 124-136. doi: doi.org/10.1016/j.isprsjprs.2019.08.018

Stickney, W. (2024). Cessna 210N Price and Operating Costs. https://boltflight.com/cessna-210n-price-and-operating-costs/.

Taylor, J. (1982). Jane's all the world's aircraft, 1982-83: Seventy-third year of issue. Jane's Publishing Company. Retrieved from https://books.google.nl/books?id=prK3GAAACAAJ

UN GGIM. (2015). Future trends in geospatial information management: the five to ten year vision (second edition). United Nations Committee of Experts on Global Geospatial Information Management. Retrieved from https://ggim.un.org/future-trends/

United Nations Conference on Trade and Development. (2023). Review of maritime transport 2023 (2023rd ed.). United Nations. doi: doi.org/10.18356/9789213584569

United Nations Economic Commission for Europe. (2024). Codes for Trade | UNECE. Retrieved 2024-11-15, from https://unece.org/trade/cefact/UNLOCODE-Download

UNODC. (2022). World Drug Report 2022. United Nations publication. Retrieved from https://www.unodc.org/unodc/data-and-analysis/world-drug-report-2022.html (United Nations Office on Drugs and Crime)

van Schilt, I. M., Kwakkel, J. H., Mense, J. P., & Verbraeck, A. (2024a). Dimensions of data sparseness and their effect on supply chain visibility. Computers & Industrial Engineering, 191, 110108. doi: 10.1016/j.cie.2024.110108

van Schilt, I. M., Kwakkel, J. H., Mense, J. P., & Verbraeck, A. (2024b). Identifying the structure of illicit supply chains with sparse data: A simulation model calibration approach. Advanced Engineering Informatics, 62, 102926. Retrieved from https://www.sciencedirect.com/science/article/pii/S1474034624005779 doi: doi.org/10.1016/j.aei.2024.102926

Varga, L., Kovács, A., Geza, T., Papp, I., & Néda, Z. (2016). Further we travel the faster we go. PloS one, 11, e0148913. doi: 10.1371/journal.pone.0148913

Vermeulen, I., Van der Leest, W., & Dirksen, V. (2018). De doorvoer van cocaïne via Nederland (Tech. Rep.). Zoetermeer: Politie Dienst Landelijke Informatieorganisatie.

Wilches, F. J., Burbano, J. L. A., & Sierra, E. E. C. (2020). Vehicle operating speeds in southwestern colombia: An important database for the future implementation of optimization models for geometric design of roads in mountain topography. Data in Brief, 32, 106210. Retrieved from https://www.sciencedirect.com/science/article/pii/S2352340920311045 doi: doi.org/10.1016/j.dib.2020.106210

Zhang, L., Zhang, L., & 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. doi: 10.1109/MGRS.2016.2540798

A | Literature on route identification and machine learning methods

Paper	Main objective	Method
1. Robert et al. (2015)	Potential surface analysis of an area for smuggling	Analytic Hierarchy Process, a multi criteria decision making analysis. Criteria based on surface area, socio-economic factors and narcotic factors (presence of police and production sites).
2. Magliocca et al. (2019)	Analyse the effect of policies on smuggling	Agent Based Model, where agents choose paths based on policies and land characteristics.
3. Magliocca et al. (2022)	Estimate the suitable of an area for smuggling	Mixed effects model, where policy and land characteristics are included in the effects.
4. Achi et al. (2012)	Predict optimal routing for illicit waste dumping	Selection of illicit destination based on land characteristics. This was combined with the official road network. Then calculated the shortest path based on distance.
5. Niamkaeo & Robert (2020)	Estimate smuggling risks for areas based on open source data	Combining news data on drug seizures with land use data. Tree-based classification algorithm to estimate smuggling risk for each area.

Table A.1: Literature on models to identify routes for illicit activity

Paper	**Main objective**	**Method**	**Data**
1. Bolívar-Santamaría & Reu (2021)	Identifying agroforestry systems	Random Forest regression model	Sentinel-2
2. Escobar-López et al. (2022)	Identifying coffee-agroforestry systems	Random Forest regression model	Sentinel-2
3. Nazarova et al. (2020)	Detecting vegetation change in presence of high cloud cover	Artificial Neural Network	Sentinel-2
4. Shendryk et al. (2019)	Multi-modal classification of cloud and land cover	Convolutional Neural Network	Sentinel-2 and PlanetScope
5. González-González et al. (2022)	Land use-land cover classification	Random Forest and Support Vector Machine	Landsat-8
6. Berkson et al. (2020)	Comparison of methods to detect and segment illicit coca fields	Traditional methods (e.g. maximum likelihood classification and SVM) and Convolutional Neural Networks	Sentinel-2
7. Pinto Hidalgo & Silva Centeno (2022)	Detecting potential illicit laboratories	Convolutional Neural Network	PlanetScope
8. Becerra et al. (2021)	Detecting illicit ships and airstrips	Convolutional Neural Network	Sentinel-1
9. Botelho et al. (2022)	Detecting unofficial roads	Convolutional Neural Network	PlanetScope

Table A.2: Literature on machine learning models identifying (illicit) cultivation and infrastructure

B | Supporting figures to case study

Figure B.1 shows an example of the routes from fields to laboratories, where there is a slight difference in choice of route. Where cost and travel time are mostly similar, with the weight of risk, there is in the middle of the graph a path through a waterway to a laboratory that is not in the graphs above. It is likely due to it being a route with not as many edges, but is longer in distance and therefore in cost and travel time.

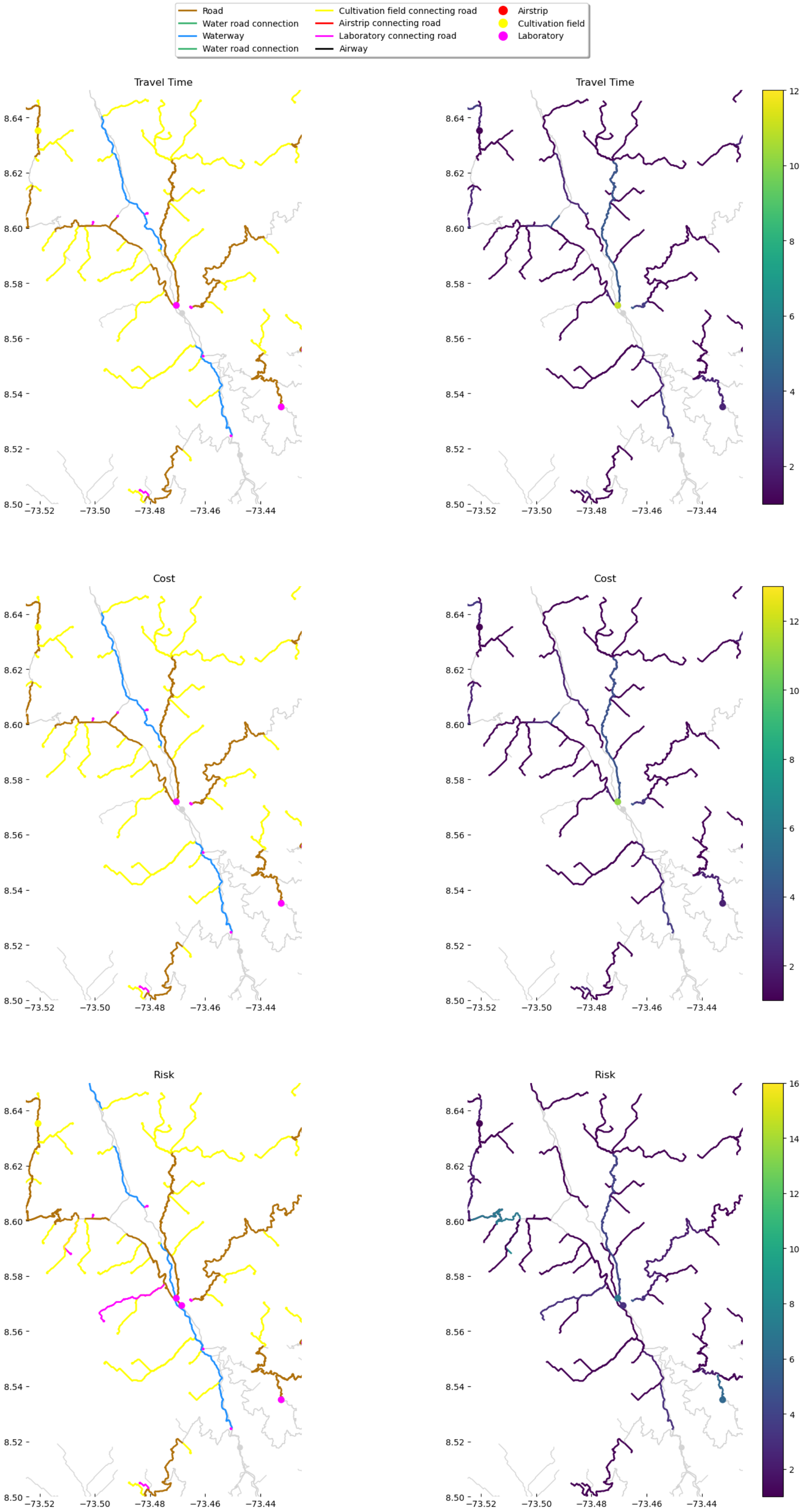


Figure B.1: Close-up of graph with routes and edge frequencies: cultivation fields to laboratories

Figure B.2 show a close-up of the smaller case study area for the routes from laboratories with ports. In this close up it is clear that the edge frequencies differ for each weight metric. The risk metric differs the most from the other metric in terms of travelled edges.

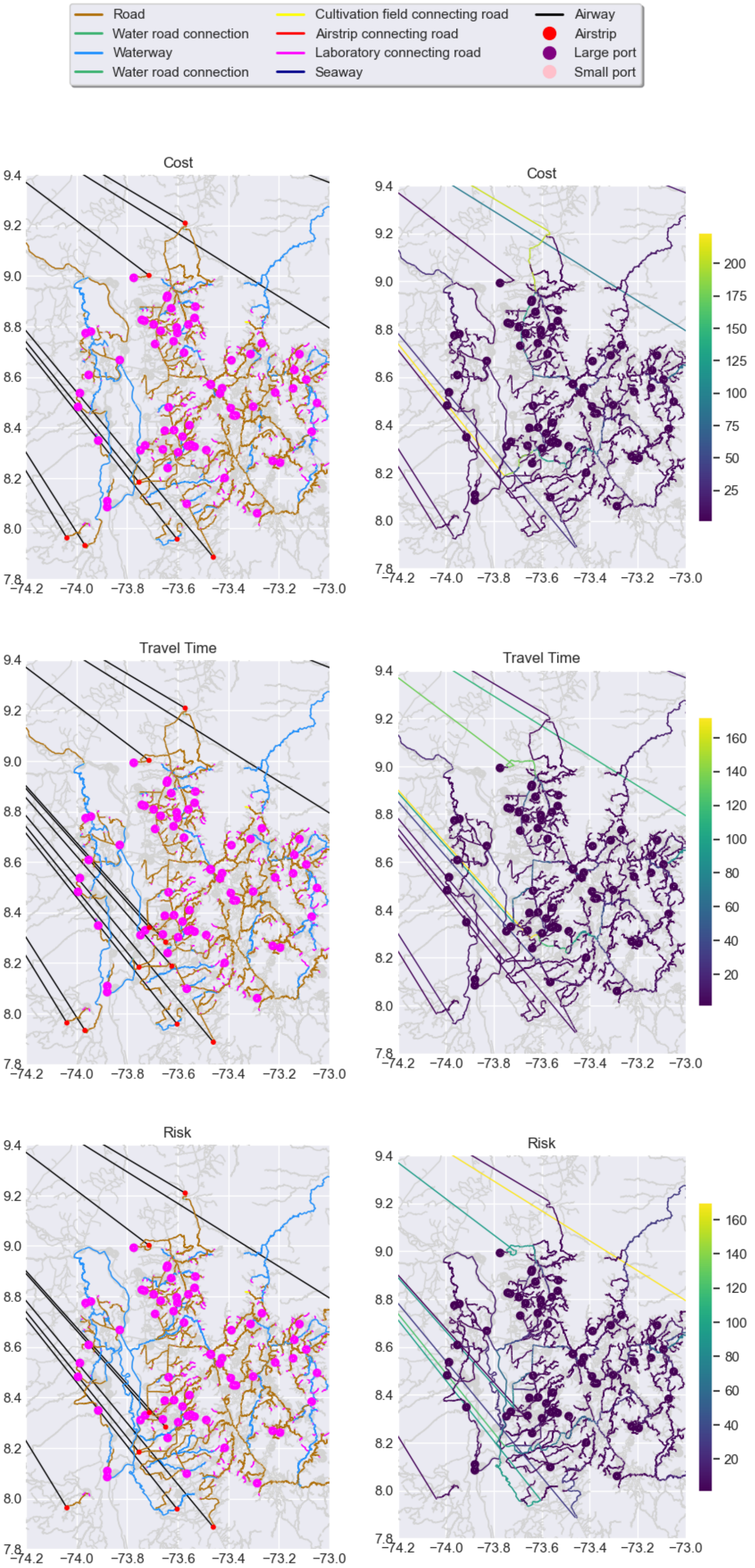


Figure B.2: Close-up of graph with routes and edge frequencies: cultivation laboratories to port