UNIVERSITÀ DEGLI STUDI DI TRIESTE

XXXVIII CICLO DEL DOTTORATO DI RICERCA IN
APPLIED DATA SCIENCE AND ARTIFICIAL INTELLIGENCE

Istituto Nazionale di Oceanografia e di Geofisica Sperimentale (OGS)

MACHINE LEARNING FOR AUTOMATED RECOGNITION OF SEABED MORPHOLOGIES

Settore scientifico-disciplinare: INF/01

DOTTORANDO
UMBERTO DI LAUDO

COORDINATORE
PROF. FRANCESCO PAULI

SUPERVISORE DI TESI
PROF. LUCA MANZONI

SUPERVISORE DI TESI
DOTT.SSA SILVIA CERAMICOLA

ANNO ACCADEMICO 2024/2025

Università degli Studi di Trieste

Ph.D. in Applied Data Science & Artificial Intelligence

XXXVIII cycle

Machine Learning for Automated Recognition of Seabed Morphologies

Candidate
Umberto Di Laudo

Supervisors
Prof. Luca Manzoni
Dr. Silvia Ceramicola

Summary

The seabed hosts a wide variety of morphological features that record the geological and geomorphological processes shaping continental margins. Their systematic recognition is fundamental for marine geohazard assessment and for the planning of offshore infrastructures. However, conventional mapping relies heavily on manual interpretation of bathymetric data, which is time-consuming, subjective, and difficult to reproduce consistently across large areas. This thesis explores the use of machine learning for the automated recognition of seabed morphological elements from high-resolution bathymetric derivatives. A convolutional architecture based on U-Net is developed and applied to a dataset from the MaGIC Project, covering selected regions of the Italian continental margins. The proposed framework includes dedicated preprocessing, data balancing, and a bidirectional neighborhood-based metric specifically designed to evaluate elongated and discontinuous structures that are poorly captured by standard pixel-wise scores.

Three model configurations are tested, multi-class (reduced-class, and binary) to analyze how class granularity affects performance and interpretability. Overall, the proposed framework bridges quantitative image segmentation with geological interpretation, demonstrating that deep learning can effectively support and accelerate seabed mapping, while preserving consistency and reproducibility across extensive marine domains.

Contents

Chapter 1

Introduction

1.1 Motivations and Challenges

The seabed displays a remarkable variety of geomorphological structures, including ridges, basins, canyons, escarpments, sediment waves, and landslide scarps. Each of these features is composed of smaller structural components called *morphological elements*, that together define its shape and organization. Understanding these components is fundamental to reconstructing the geological and morphological evolution of the seafloor. Mapping the seabed plays a crucial role in marine geoscience, as it supports both the assessment of marine geohazards and the planning of offshore infrastructures, such as pipelines, wind farms, and submarine cables [1, 2]. Nevertheless, traditional mapping relies largely on expert interpretation of bathymetric data, a process that is accurate but slow, subjective, and difficult to reproduce on a large scale.

As bathymetric data become more detailed and cover larger areas, keeping manual interpretations consistent across wide regions becomes increasingly difficult. This issue was particularly evident in the MaGIC Project (Marine Geohazards along the Italian Coasts) [3], which produced the first detailed atlas of the Italian continental margins. During the MaGIC project geologists generated a valuable labeled dataset that now provides an ideal foundation for developing and testing automated classification methods.

Within the MaGIC framework, many seabed morphological elements are represented by thin, elongated, and discontinuous lineaments rather than compact structures, making both annotation and automatic detection more challenging. These lineament-like elements constitute the main focus of this thesis. The dataset is strongly unbalanced, with a dominant background class and a few rare morphological elements. Some categories, such as “Generic Escarpment”, are intentionally broad or ambiguous, reflecting the interpretative nature of the labeling process. These characteristics make the problem of automated classification both scientifically interesting and technically challenging.

In this context, Machine Learning (ML) provides a powerful complementary approach.

Convolutional encoder–decoder architectures [4, 5], such as U-Net [6], can learn from expert-labeled examples and generate reproducible preliminary maps from bathymetry-derived inputs, including depth, slope, and curvature. When properly designed, these models can capture spatial patterns that correspond to morphological elements and thus support geologists in the interpretation process. However, applying ML to seabed morphology raises specific issues. The elements of interest are sparse and elongated, often forming discontinuous lineaments that are difficult for the model to represent effectively. The dataset's imbalance can bias training toward the most common classes. Expert annotations are also inherently uncertain, since they reflect interpretative choices rather than strictly measurable boundaries. Finally, evaluating predictions requires metrics consistent with geological reasoning: traditional pixel-wise measures, such as Cross Entropy or Dice, may over-penalize small spatial shifts that are irrelevant from a geomorphological standpoint.

Accordingly, this thesis proposes a segmentation framework designed to address these challenges. It adopts a patch-based training strategy in which training and test areas are spatially separated to ensure fair evaluation. Before training, overlapping windows are used to increase sample diversity, and a simple thickening operation makes thin lineaments easier for the model to learn and recognize. After prediction, a light post-processing step removes small isolated regions that are unlikely to correspond to real elements. The evaluation combines standard pixel-based metrics with a *neighborhood-based, bidirectional* measure that better reflects the spatial proximity between predictions and expert interpretations. Beyond the full multi-class setting, the work also explores alternative *groupings* of classes, aggregating morphologically related elements, and an *extreme binary* formulation (lineament vs. background) to examine how class granularity influences robustness and practical usefulness.

1.2 Research Questions

This thesis investigates how far ML can assist expert-based seabed interpretation from bathymetry-derived products alone, and under which conditions it is most reliable. The guiding questions are:

- **Model adaptation:** How can a U-Net–based pipeline be adapted to detect *thin, elongated* morphological elements from depth, slope, and curvature maps, using expert interpretations as supervision?
- **Label design and class granularity:** How should seabed features be *grouped* for stable and useful classification? What changes when moving from a full multi-class setup to *aggregated* classes (e.g., escarpments/scarps merged) and to a *binary* detector (lineament vs. background)?

- **Training protocol:** What is the role of *patching with overlap* and spatial separation of areas in shaping generalization? Which simple *pre-/post-processing* choices (line-thickening, blob cleaning) meaningfully affect the outputs?

- **Evaluation aligned to practice:** How can model performance be assessed through *tolerance-aware metrics* that account for spatial proximity and continuity of lineaments, providing a more meaningful measure than standard pixel-wise scores under interpretative uncertainty?

- **Data-conditional behavior and scope:** How does the model's reliability vary as a function of the input geomorphic variables (depth, slope, curvature), and what do these dependencies suggest about its potential generalization to regions with different morphodynamic regimes?

To address these questions, this work develops a U-Net segmentation model that uses depth, slope, and curvature as input data. Different groupings of classes are tested, from the complete set of morphological elements to simplified versions where similar features are merged, and to a binary case that separates lineaments from the background. The model is trained and evaluated using a patch-based approach with overlapping windows to increase sample diversity and ensure a fair comparison between areas. Its performance is then assessed through both standard pixel-based metrics and a bidirectional, neighborhood-based measure that better reflects the way geologists compare predicted and real features. Finally, the study includes an analysis of how model performance changes with the main geomorphic variables (depth, slope, and curvature) to understand under which conditions the approach works best. The goal is to show not only what has been done, but also how each design choice affects the clarity and practical usefulness of the results, without attempting to replace expert interpretation.

1.3 Structure of the Thesis

The manuscript is organized to move from context to implementation, and then to analysis and implications:

- **Chapter 2 – State of the Art:** reviews the methods for seabed data acquisition, the main morphological features of interest, and the evolution from statistical to ML and Deep Learning (DL) approaches in marine geology. It also discusses foundations of Convolutional Neural Networks (CNNs), image segmentation, and U-Net.

- **Chapter 3 – Problem Statement and Dataset:** study area and inputs (depth, slope, curvature); expert interpretations and label grouping; dataset characteristics (sparsity, imbalance); pre-processing of annotations (thickening).

- **Chapter 4 – Methodology:** U-Net architecture and training protocol (patching with overlap); pipeline details (input normalization, pre-/post-processing, evaluation); definition of a neighborhood-based metric.
- **Chapter 5 – Experiments and Results:** *experiments across class granularities* (multi-class, reduced-class, binary) with qualitative/quantitative analyses guided by the proposed metrics.
- **Chapter 6 – Conclusions and Future Work:** synthesis of findings with respect to the research questions; limitations tied to sensing, labels, and metrics; open directions, including task-aligned evaluation, data fusion, imbalance handling, and geohazard-oriented products.

1.4 Publications

During the course of this Ph.D., the following publication was produced and presented at a national conference:

- U. Di Laudo, S. Ceramicola, and L. Manzoni, *Machine Learning for Automated Seabed Mapping*, in *Ital-IA 2024 – Thematic Workshops*, *CEUR Workshop Proceedings*, vol. 3762, 2024.
  `https://hdl.handle.net/11368/3118040`

Part I

Context and Background

Chapter 2

Background and Related Works

In this section, we present the existing state of the art in the area of mapping the seabed and in image segmentation. As will be shown, although some works apply Deep Learning (DL) techniques to marine geology, no automatic method currently exists for seabed mapping via segmentation that captures distinct morphological lineaments, highlighting a gap our work aims to address.

In Section 2.1, we describe the current techniques to obtain morphological data, like bathymetry maps. Section 2.2 gives an introduction on what is a *morphological features*, and in which way it is usually represented. In Section 2.3, we provide a comprehensive overview of the different DL approaches that have been employed in the field of marine geology, with a particular focus on their application to seabed mapping. Section 2.4 reviews the relevant literature on CNNs and image segmentation with a particular focus on U-Net architecture which is the primary technique used in this work.

2.1 Seabed Data

In marine geology field obtain data is often an hard and expensive work. In fact, geological analysis are usually done using underwater data, obtained through the use of underwater acoustic systems emerged in the early part of the 20th century [7]. The earliest and the most used system developed for these goals is the single-beam bathymetric sounders[7]. Single-beam echo-sounders, often referred to as acoustic ground-discrimination systems, are employed to investigate the reflective properties of the seafloor. These instruments emit sound pulses, typically in the 30–200 kHz range, which bounce back from the seabed and are recorded by a transducer [8, 9]. Knowing the propagation speed of sound c and the time of flight t, the distance d is calculated as [10]

$$d = \frac{c \times t}{2} \tag{2.1}$$

More advanced Advanced Wire Delivery Systems (AGDS) technologies have since

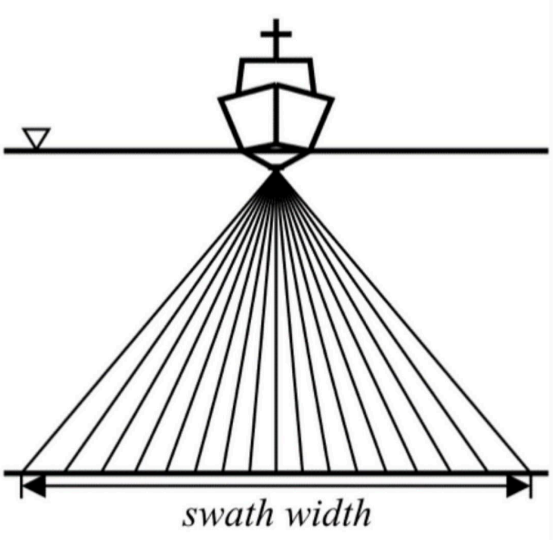


Figure 2.1: From [10]. A simple schematic illustration of a multibeam echosounder system.

been designed to analyze the seabed's acoustic reflectance characteristics. The character of the returned signal is not determined solely by sediment grain size, sorting, density, porosity, or small-scale topographic features, but can also be affected by the abundance, type, and distribution of benthic organisms and vegetation[8, 9]. The key advantage of this sensor lies in its ease of use and relatively low cost compared to other sensors. However, its main limitation is the narrow coverage area (typically ranging from 2° to 12°), which restricts its ability to capture wide coverage [10].

The evolution of this instrument is the Multibeam echosounder systems (MBES) [11], advanced sonar systems that emit a fan-shaped array of acoustic pulses to simultaneously collect hundreds of depth measurements across a wide swath of the seafloor. A simple representation of a multibeam echosounder system is shown in Figure 2.1. The first system appeared in 1979 with a coverage angle of 45° and forming 16° beams [11, 12]. This instrument is mainly used to measure the depth of the seabed, computed by measuring the time taken by the sound beam to travel the distance from the ship to the seabed and back [7, 12].

Modern multibeam echo-sounder systems offer significantly enhanced angular coverage, between 120° and 150°, and can form hundreds of individual beams [7]. These systems provide extremely high-resolution data, enabling detailed and accurate mapping of seabed morphology that is key to the classification process [7]. Bathymetric data collected from a MBES are generally derived from the combined measurement of travel time and incidence angle for each transmit/receive beam. The specific calculation method can vary depending on the system's design and the beam's position within the swath [12]. In cases where the sound velocity profile in the water column is uniform, straightforward geometric computations can be applied to estimate water depth [7].

Another instrument to acquire seabed data is Sidescan sonar (SSS). SSS is a key technique for seafloor imaging, producing high-resolution, photo-like mosaics widely used in habitat mapping [8]. Its performance depends on system frequency: low frequencies penetrate deeper but provide limited resolution, while high frequencies are optimal for surface mapping and sediment discrimination. Recent advances, including "CHIRP" and synthetic aperture sonar (SAS), have improved range and resolution by using wide-bandwidth pulses and variable frequencies [8, 9]. Modern digital SSS systems, often operating in CHIRP mode, can detect objects of decimeter scale within swaths up to $\sim 200m$, offering detailed and reliable seabed characterization from shallow to deep-water environments[8, 9].

In our work we use an high-resolution bathymetry, obteined with multi-beam during the MaGIC project[3].

2.2 Seabed Morphological Elements

Seabed bathymetry represents the primary source of information for understanding the morphology and composition of the underwater environment. By analyzing high-resolution bathymetric data, marine geologists identify, delineate, and classify the main morphological features that shape the seafloor. These features include ridges, basins, canyons, escarpments, and channels, each defined by its own geometry and sedimentary structure.

Within these large-scale features, smaller structural components, known as morphological elements, form the essential building blocks of seabed morphology. Examples include canyon walls, slope breaks, and scarps, which together define the internal architecture of larger geomorphic units. The identification and characterization of such elements represent the first step in seabed interpretation and mapping.

Beyond its scientific relevance, seabed mapping has significant practical implications. It provides the basis for assessing marine geohazards, such as submarine landslides or fault-related deformations, and for designing and locating offshore infrastructures. Understanding the relationships between geomorphological patterns and geological processes is therefore a key objective in marine geology [1, 2].

Marine geohazards comprise a wide range of processes and events linked both to contemporary morphodynamic activity and to the long-term geological evolution of continental margins. While the geological mechanisms shaping these margins have long been a key focus of marine geology, their associated geohazard implications have often been treated as a secondary, though important, aspect. In recent years, significant advances in seafloor imaging and mapping technologies have spurred dedicated research on marine geohazards. These technological improvements are reshaping investigative strategies in marine geoscience, with detailed mapping of seafloor geomorphology now serving as a primary step for both hazard identification and the study of underlying geological processes and environments [2]. Geohazard-related features often originate from the mobilisation and reshaping of the seafloor driven by various processes. Intense erosion, sedimentation, and mass-wasting events are among the primary factors shaping submarine landscapes, frequently resulting in prominent structures such as submarine canyons. Tectonic activity, including active faults, can deform the seafloor directly or from buried structures, while deep-seated deformation may cause fluid overpressure, leading to the formation of pockmarks, fluid escape features, or mud volcanoes. These processes not only alter seafloor morphology but also disrupt sediment stratigraphy and weaken its mechanical stability, increasing susceptibility to failure. Submarine landslides, which vary widely in scale and mechanism, may generate distal mass flows or turbidity currents. Additionally, strong bottom currents contribute to continuous seafloor remodelling by transporting sediments and forming bedforms interspersed with erosional zones [2].

In this framework italian project MaGIC (Marine Geohazards along the Italian Coasts) was fundamental, which produced the first comprehensive atlas of the seabed of the Italian continental margins [3]. The project consisted of two main parts: the acquisition of an high-resolution multibeam bathymetry, covering the italian margins and the interpretation of that data, in order to create the map of geohazard features of the italian seas composed by 72 Sheets [2]. In that project, each of these sheet, are made up by four cartographic levels:

- Level 1: Physiographic domains representing the geological and physiographic setting of the sheet
- Level 2: Morphological Units
- Level 3: Morphological Elements
- Level 4: Critical Zones

Our work is mainly focussed on the Level 3, i.e. on *Morphological Elements* such as basins, canyons, and channels. In the MaGIC project these elements are represented as lineaments [3].

2.3 Machine Learning Techniques in Marine Geology

In recent years, Machine Learning has become increasingly popular in marine geology, expanding the range of available approaches and demonstrating performance comparable to, and sometimes surpassing, traditional manual techniques [13, 14, 15]. This growing adoption is also motivated by the limitations of conventional sediment sampling and manual interpretation, which are labor-intensive and constrained in spatial coverage [10]. In contrast, ML leverages the growing amount of acoustic, optical, and hyperspectral data collected through autonomous and remotely operated platforms, offering new possibilities for mapping and characterizing the seabed.

2.3.1 From statistical to machine learning approaches

Early work in automated seafloor classification relied mainly on statistical frameworks. Bayesian classification methods were applied to MBES backscatter data, sometimes at multiple frequencies, to discriminate sediment classes with improved accuracy [16, 17, 18, 19, 20]. These techniques demonstrated robustness in controlled scenarios but were limited by assumptions such as Gaussian-distributed intensities, which do not always hold in heterogeneous environments [10].

Building on these foundations, ML algorithms were increasingly tested for seabed mapping. Methods such as Support Vector Machines (SVM), Random Forests (RF), k-Nearest Neighbors (kNN), and Back-Propagation Neural Networks (BPNN) have been widely used with acoustic and optical data [21, 22, 23, 24]. For example, Lawson et al. [21] classified noisy seamount features from bathymetry using RF and Extremely Randomized Trees, achieving accuracy up to 97%. Ji et al. [22] introduced the Selecting Optimal Random Forest (SORF) to improve both efficiency and accuracy in MBES sediment classification, while Zhao et al. [23] combined backscatter and Random Forest to discriminate multiple sediment types in the Belgian North Sea. Neural networks optimized with techniques like Particle Swarm Optimization and AdaBoost have also been proposed to prevent local minima and enhance classification outcomes [25]. Beyond acoustics, computer-vision-inspired ML pipelines have been tested on coral reef imagery, where texture, shape, and color features classified by SVM or ensemble models provided effective benthic habitat mapping [26, 27, 28]. Collectively, these works highlight the flexibility of classical ML, though they often require extensive manual feature engineering and preprocessing [10].

2.3.2 Object-Based Image Analysis (OBIA)

Currently, one of the most widely adopted supervised approaches in marine mapping combines object-based image analysis (OBIA) [29, 30] with ML algorithms such as decision trees, SVM, or RF. OBIA partitions raw inputs, such as side-scan sonar mosaics or digital elevation models, into objects rather than individual pixels, which are then described by morphological or textural attributes and used for classification [31]. This approach has proven effective in diverse applications, including the detection of coral mounds [32, 33, 34], the study of sediment waves [35], and benthic habitat mapping [31, 36, 37].

Several recent works have confirmed its versatility: Menandro et al. [38] applied

OBIA to side-scan sonar backscatter for reef mapping on the Brazilian continental shelf; Koop et al. [39] used bathymetric derivatives such as slope and rugosity for sediment classification; and Janowski et al. [40] combined backscatter and bathymetry to distinguish six nearshore habitats in the Baltic Sea. In many cases, OBIA has been shown to outperform or complement pixel-based ML approaches, demonstrating robustness across sensors and scales.

2.3.3 Deep learning for seafloor characterization

In recent years, alongside traditional ML methods, DL, in particular CNNs, has been increasingly explored for habitat and geomorphological mapping. DL techniques are particularly appealing because they can automatically learn complex patterns from raw data, reducing the need for expert-driven preprocessing and manual feature design while often improving classification accuracy [10]. Most DL applications so far have focused on ecological and habitat mapping [41, 42], but their scope is rapidly expanding.

CNNs have been tested on side-scan sonar datasets, where shallow CNNs outperformed deeper networks in sediment classification tasks [43]. Generative Adversarial Networks (GANs) have been used to augment limited sonar datasets, improving CNN training stability and performance [44, 45]. Artificial Neural Network (ANN)-based pipelines estimated polymetallic nodule abundance using side-scan backscatter [46, 47]. State-of-the-art object detectors such as YOLOv5 and Faster R-CNN have been applied for underwater debris detection and target recognition in sonar data, reaching mean average precision above 85% [48, 49, 50]. For optical images, Mask R-CNN pre-trained on the COCO dataset proved particularly effective in segmenting manganese nodules with high accuracy (97%), overcoming the similarity between nodules and background sediments [51].

Beyond classification, DL architectures have also been tested for segmentation and pattern recognition of seabed features. An Adaptive Neuro-Fuzzy Inference System (ANFIS) framework integrated multibeam bathymetry and side-scan sonar for semi-automatic classification of submarine lava flows [52]. Autoencoder networks have learned latent representations of seamounts in global bathymetric data, enabling systematic detection across regions [53]. ANN-based pipelines analyzed bathymetric cross-sections to discriminate fault scarps in the Arctic [54], while GANs and CNNs have been employed to detect and segment pockmarks in estuarine and continental-slope bathymetry [55]. Collectively, these studies confirm the feasibility of DL for identifying specific geomorphological features, though typically restricted to single feature types and requiring supervised or semi-supervised learning.

Among the most recent contributions, Arosio et al. [31] applied neural networks for seabed segmentation, but their method focused on large-scale regions (e.g., Plane, Ridge, Depression) rather than fine-scale, hand-drawn lineaments considered in this work. To the best of the authors' knowledge, the automatic detection of such linear features remains largely unaddressed. Figure 2.2 presents the segmentation results obtained with the ResNet50 model [56] under different input configurations. Specifically, model #30 was trained with bathymetry and hillshades, model #26 with hillshades only, and model #40 with a full set of inputs including bathymetry, hillshades, vector ruggedness measurement (VRM), and aspect (northness and eastness).

2.3.4 Towards the detection of linear features

All the referenced previous works are oriented toward the detection of area-based patterns, whereas lineament identification remains largely unexplored. One notable exception is the study by Kokinou and Panagiotakis [57], which introduced pattern recognition techniques applied to bathymetric data from the Eastern Mediterranean. Their pipeline combined filtering and skeletonization with geometric classification of segments, successfully detecting tectonic lineaments and salt-related structures that influence seafloor morphology. However, the methodology was limited to a restricted set of features and relied primarily on classical image processing. Figure 2.3 shows the result of the filtering processes (a), while (b) displays the skeleton (medial axis) plotted on the bathymetry data in the Levantine Basin (east and southwest of the Eratosthenes Seamount).

The novelty of our work lies in shifting the focus from broad geomorphological units to the detection of diverse linear features—such as generic escarpments, continental shelf scarps, and canyon scarps—through DL segmentation. By targeting these structures, we aim to deliver a clearer and more complete description of seafloor morphology, directly supporting geological interpretation, resource exploration, and geohazard assessment.

2.4 Deep Learning Foundations for Image Analysis

2.4.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a family of DL models that have become the standard tool for analyzing visual data. Their key advantage lies in their ability to automatically learn useful features directly from raw input, such as images or signals, without relying on handcrafted descriptors. This makes them particularly effective for tasks in which spatial structures and local correlations are essential, including image classification, object detection, and semantic segmentation [5].

The structure of a CNN is inspired by the visual perception [4, 58], where artificial neurons correspond to biological ones, and they respond to specific local patterns and progressively combine them to detect more complex structures. At the core of CNNs are convolutional layers, where small learnable filters (or kernels) slide across the input

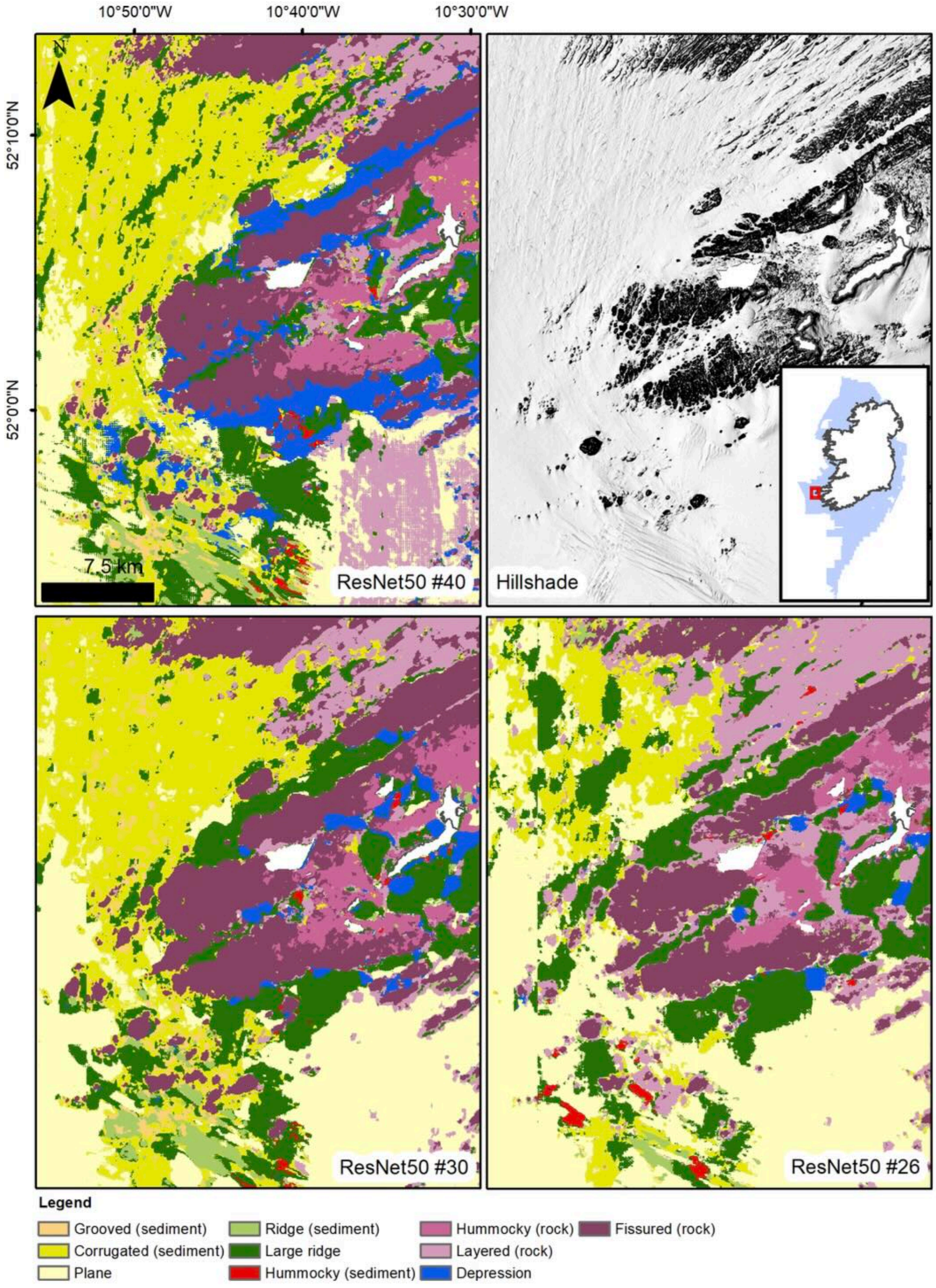


Figure 2.2: From [31]. Segmentation results of the ResNet50 models with varying input layers: model #30 (bathymetry + hillshades), model #26 (hillshades only), and model #40 (bathymetry + hillshades + VRM + aspect functions).

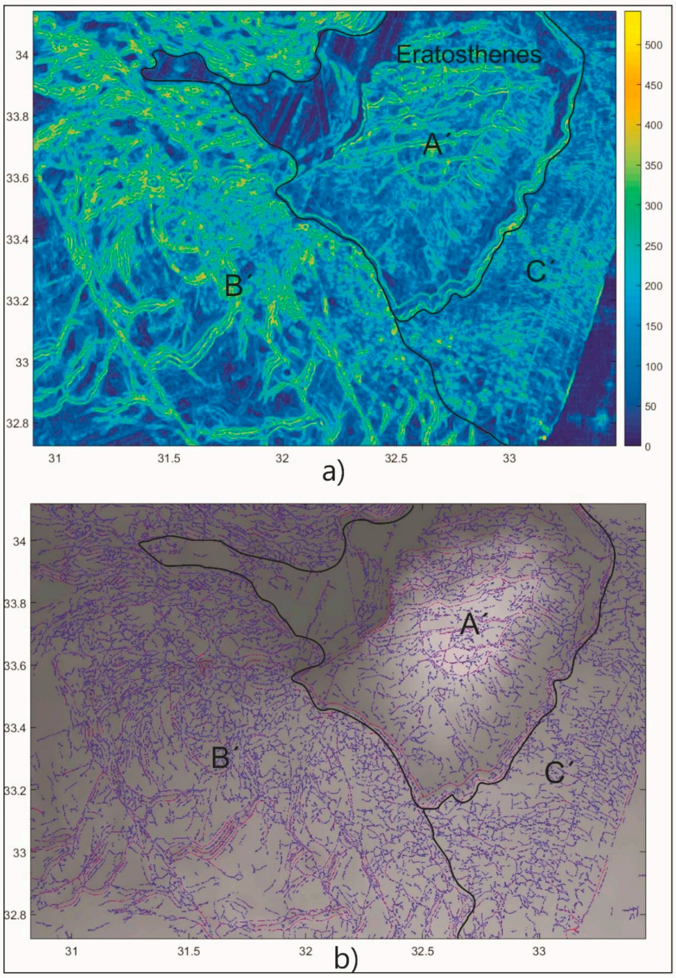


Figure 2.3: From [57]. Automatic detection of seafloor lineaments in the Levantine Basin (east and southwest of the Eratosthenes Seamount). (a) Multi-filtering output with detection strength (0–550). (b) Extracted skeletons on the bathymetric surface, where red lines mark strong and blue lines weak features.

image to extract local features such as textures, edges, or simple shapes [5]. As information flows through multiple layers, these local features are aggregated into higher-level representations, enabling the network to recognize complex objects or patterns. Pooling layers are often included to reduce spatial resolution while keeping the most relevant information, improving both efficiency and robustness to translations. Non-linear activation functions, such as the Rectified Linear Unit (ReLU) [5, 59], make the network capable of modeling complex relationships. Finally, fully connected layers combine the learned features for prediction tasks, such as assigning class labels [4, 5, 59].

Over the years, CNNs have evolved through the development of increasingly deep and specialized architectures. LeNet, proposed by LeCun et al. [60], was one of the first successful CNNs, designed for digit recognition. Later, AlexNet was introducted. It use convolutional filters larger than those employed in LeNet, integrate local response normalization (LRN) and adopt overlapping pooling layers. VGG networks showed that stacking small convolutional filters can improve performance, while ResNet introduced residual connections, enabling the training of very deep models without degradation. Other architectures, such as Inception or MobileNet, further optimized efficiency and scalability [5].

The success of CNNs is strongly tied to their ability to capture spatial hierarchies, making them highly suitable for image segmentation. In this context, CNNs are often used as the backbone of encoder–decoder architectures (e.g., U-Net), which are specifically designed to preserve spatial detail while learning abstract feature maps. These models have set the state of the art in many segmentation tasks and represent the foundation for the approaches adopted in this thesis.

2.4.2 Image Segmentation

Image segmentation is a computer vision technique that partitions a digital image into multiple segments or regions.

According to [61], image segmentation methods can be grouped into three categories: semantic segmentation, instance segmentation, and panoptic segmentation. In image segmentation, objects in an image are often categorized into *things* and *stuff* [61]. *Things* refer to countable objects with well defined shapes and boundaries, such as cars, people, or chairs, while *stuff* denotes uncountable regions with homogeneous appearance and no distinct instances, such as sky, road, or grass. *Semantic segmentation* classifies each pixel into a category without distinguishing between individual *things* of the same class [62]. For example, in a figure containing several cars, all car pixels are labeled as "car", but the individual vehicles are not distinguished. *Instance segmentation* builds on semantic segmentation by identifying each object instance separately [63]. In the same example, each car is recognized as a distinct object, though non-countable elements i.e. *stuff* like the road are ignored. *Panoptic segmentation* unifies both approaches by as-

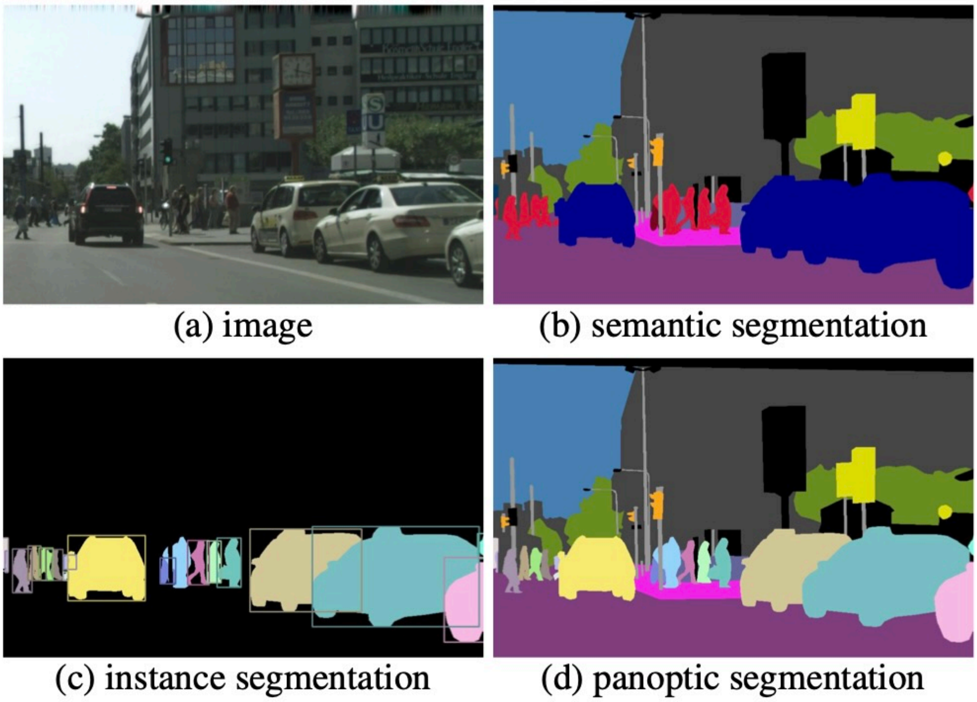


Figure 2.4: Figure from [61]. Example comparison of image segmentation approaches. (a) Original street scene. (b) Semantic segmentation: all pixels are assigned a class label, with no distinction among different object instances of the same class (e.g., all cars in blue). (c) Instance segmentation: each individual object instance (*thing*) is separated, while *stuff* classes like the road are typically not considered. (d) Panoptic segmentation: integrates both views, assigning unique labels to all pixels, distinguishing between multiple *thing* instances (cars, people) and *stuff* classes (road, sky, building).

signing a unique label to each region, whether it corresponds to an object instance or a background class [64].

This concent is well illustrated in Figure 2.4 from [61] where for the same picture (a), all three segmentation approach are applied. As it easy to see, semantic segmentation classified all pixels (both *things* and *stuff*) of the image in a specific class, but countable elements of the same type are not distinguishable; on the other hand instance segmentation (c) distinguishes each individual belonging to the same classes, but it ignores *stuff*; finally panoptic segmentation (d) unifies these two approaches, in such a way that both *thing* and *stuff* are detected.

Before the rise of DL, image segmentation relied on a wide variety of region-based and edge or boundary-Based techniques. These methods are typically categorized into approaches exploiting similarity (grouping pixels with common properties) and those based on discontinuity (detecting boundaries and edges). Since no universal solution exists, segmentation methods are often tailored to the application at hand [65].

Region-based techniques divide an image into homogeneous regions by clustering pixels that share similar attributes such as intensity, color, or texture. According to Zaitoun and Aqel [65] typical region-based techniques are:

- Clustering methods: The most well-known is k-means, which partitions pixels into k clusters by minimizing the distance between each pixel and the cluster centroid. Variants include fuzzy clustering, where a pixel can belong to multiple clusters with different membership levels. While effective in medical imaging and security applications, clustering may produce discontinuous regions unless combined with other strategies [66, 67].
- Split-and-Merge: This approach begins by treating the whole image as one region and recursively splitting it into smaller subregions using quadtree structures. Adjacent regions with similar properties are then merged. The method guarantees connectivity but may suffer from blocky segmentation or over-segmentation, which can be alleviated with normalized cuts [68].
- Normalized Cuts: Based on graph theory, this method represents pixels as nodes and similarities as weighted edges. The graph is optimally partitioned by minimizing a "cut" cost that balances similarity within regions and dissimilarity across regions. It provides accurate boundary detection and integrates intensity, color, and texture in a unified framework, though it is computationally demanding [69, 70].
- Region Growing: Starting from seed points, pixels are added to a region if they meet similarity criteria. This technique is intuitive, ensures connected regions, and works well with low-noise images. However, it can lead to over-segmentation in noisy or textured images [71].
- Thresholding: One of the oldest techniques, thresholding separates foreground from background by intensity values. Global thresholding applies a single threshold across the image, while local (adaptive) thresholding adjusts values based on neighborhoods. Variants include histogram-dependent methods, P-tile techniques, and fuzzy thresholding. Thresholding is simple and fast but struggles with images where object and background intensities overlap [72].

Edge or boundary-based approaches locate object boundaries by detecting discontinuities in intensity or texture. The main techniques are [65]:

- Classical operators: Roberts, Prewitt, and Sobel are gradient-based operators that estimate edge strength and orientation by computing local derivatives. They differ mainly in kernel size and sensitivity to noise. Although effective in simple scenarios, they often require preprocessing such as noise filtering [73].
- Soft computing approaches: More sophisticated strategies include fuzzy logic, which assigns pixels to multiple sets with varying membership grades; genetic algorithms,

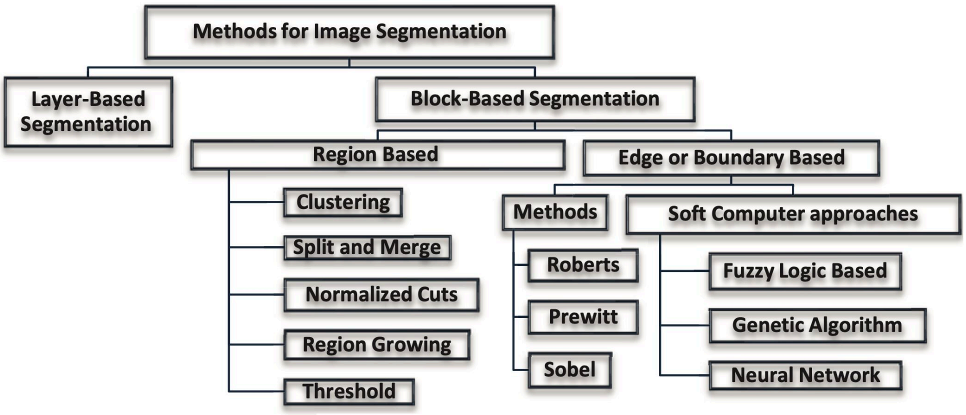


Figure 2.5: From [65]. Schematic diagram summarizing the principal image segmentation methods.

which optimize edge detection using evolutionary principles; and artificial neural networks, which learn edge features from training data. These methods improve flexibility and robustness but increase computational complexity [65, 74].

A summary diagram about all methods described above is reported in Figure 2.5 from [65].

To overcome limitations, hybrid approaches combine region-based and edge-based techniques [65]. For example, watershed algorithms or fuzzy clustering with edge constraints aim to ensure both spatial coherence and accurate boundary localization. Hybrid models are particularly useful for complex images with heterogeneous textures or illumination conditions.

The field of image segmentation has undergone a radical transformation with the advent of DL. Traditional approaches, largely dependent on handcrafted features, have given way to convolutional and encoder–decoder architectures that learn end-to-end from data [67].

The first decisive step was the introduction of fully convolutional networks (FCNs) [75], which replaced dense layers with convolutions and upsampling, making dense predictions possible. FCNs laid the foundation, but struggled with fine details and global context. Subsequent designs such as U-Net by Ronneberger et al. [6] and SegNet by Badrinarayanan et al. [76], introduced skip connections and encoder–decoder structures [59, 76] to better preserve boundaries, while later works like HRNet [77] maintained high-resolution features throughout. Over the years, several adaptations of the *U-Net* architecture have been proposed to suit different types of data. For istance, Zhou et al. [78] proposed a nested U-Net, while Çiçek et al. [79] extended U-Net to 3D images. Nowadays U-Net is also widely applied beyond biomedical contexts, including road segmentation [80], face detection [81], and autonomous driving [82].

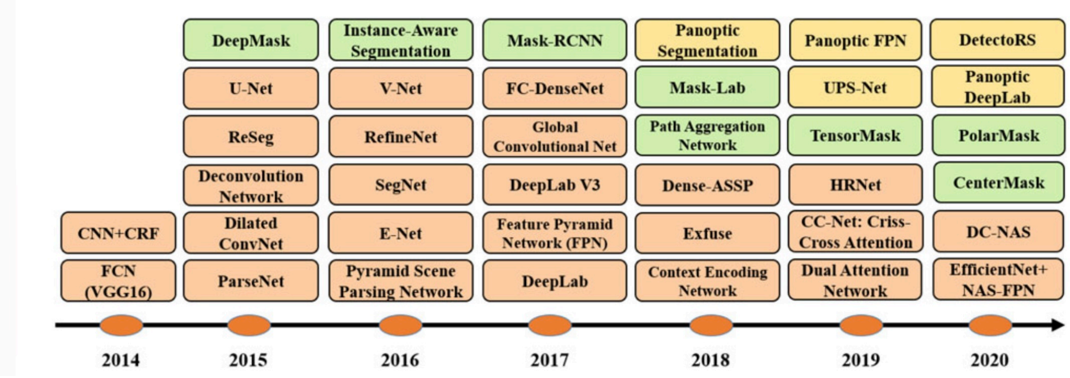


Figure 2.6: From [67]. Timeline of DL–based image segmentation methods. Approaches focusing on semantic, instance, and panoptic segmentation are highlighted in orange, green, and yellow, respectively

Early CNN-based segmentation methods often integrated probabilistic models such as CRFs to refine object boundaries and produce sharper results [83]. Subsequently, efforts shifted toward better context modeling. DeepLab popularized dilated convolutions and spatial pyramid pooling, combining wide receptive fields with high-resolution features [65, 84]. PSPNet further aggregated information from multiple spatial scales to capture global context [85]. Attention mechanisms later extended these ideas, enabling each pixel to leverage information from the entire image, as demonstrated in dual-attention [86] and OCRNet [77].

Beyond semantic segmentation, instance segmentation was revolutionized by Mask R-CNN [87], which extended object detection pipelines with a mask prediction branch, setting a strong baseline later improved by methods such as PANet [88].

For panoptic segmentation, unified frameworks such as UPSNet [89] and Panoptic-DeepLab [90] have emerged.

Overall, the state of the art is characterized by models that combine multiscale context aggregation, boundary refinement, and increasingly attention-based mechanisms. While performance continues to improve on standard benchmarks, challenges remain in terms of efficiency, interpretability, and reproducibility, highlighting the gap between academic success and real-world deployment [67].

Figure 2.6 from [67] show a timeline of the image segmentation methods that use DL.

Evaluation Metrics for Image Segmentation

The evaluation of DL models for image segmentation requires appropriate metrics that can capture not only pixel-wise accuracy but also the quality of object delineation and class-specific predictions. Although several qualitative aspects such as visual sharpness, inference speed, and memory footprint are important for real-world deployment, most of the research community has traditionally focused on quantitative accuracy measures [67]. This section reviews the most widely adopted metrics.

Pixel accuracy measures the proportion of correctly classified pixels relative to the total number of pixels in the image. For $K + 1$ classes (including K object classes and one background class), it can be expressed as:

$$\text{PA} = \frac{\sum_{i=0}^{K} p_{ii}}{\sum_{i=0}^{K} \sum_{j=0}^{K} p_{ij}}, \tag{2.2}$$

where p_{ij} denotes the number of pixels belonging to ground-truth class i that are predicted as class j. This metric is straightforward but biased toward dominant classes when the dataset is imbalanced.

To reduce the influence of class imbalance, the *mean pixel accuracy* averages the per-class accuracies across all classes:

$$\text{MPA} = \frac{1}{K+1} \sum_{i=0}^{K} \frac{p_{ii}}{\sum_{j=0}^{K} p_{ij}}. \tag{2.3}$$

Unlike simple pixel accuracy, MPA ensures that smaller classes (e.g., pedestrians in road scenes) contribute equally to the overall evaluation.

Intersection over Union, also referred to as the *Jaccard Index*, has become the standard metric for segmentation. It measures the overlap between the predicted region A and the ground-truth region B:

$$\text{IoU}(A, B) = \frac{|A \cap B|}{|A \cup B|}. \tag{2.4}$$

The metric ranges between 0 and 1, where higher values indicate a greater degree of overlap. The mean-IoU (mIoU) is obtained by averaging across all classes, providing a balanced measure of performance [67]. For semantic segmentation, mean Intersection over Union (mIoU) is become the most widely used measure, which balances accuracy across classes, while pixel accuracy and mean pixel accuracy are sometimes reported as complementary indicators.

Pixel-level classification can also be evaluated using *precision* and *recall*, defined as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{2.5}$$

where TP, FP, and FN denote true positives, false positives, and false negatives, respectively. The $F1$-score, given by the harmonic mean of precision and recall, is expressed as:

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{2.6}$$

This measure is particularly useful when false positives and false negatives carry different significance depending on the application domain.

Originally popularized in medical image analysis, the *Dice coefficient* evaluates the similarity between two samples. It is mathematically related to the F1-score:

$$\text{Dice}(A, B) = \frac{2|A \cap B|}{|A| + |B|}. \tag{2.7}$$

When applied to binary segmentation with the foreground as the positive class, the *Dice coefficient* reduces to:

$$\text{Dice} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} = F1. \tag{2.8}$$

Dice is particularly robust in cases of high class imbalance, which is frequent in biomedical imaging scenarios.

In summary, while pixel accuracy provides a coarse measurement, metrics such as IoU and Dice are more sensitive to boundary alignment and class imbalance. Consequently, they have become the dominant measures in modern benchmarks, including PASCAL VOC, Cityscapes, and MS COCO [67].

For instance segmentation, evaluation typically follows the conventions of object detection, relying on Average Precision (AP) over multiple IoU thresholds as in the COCO benchmark. Finally, the introduction of panoptic segmentation has led to the adoption of Panoptic Quality (PQ), a unified metric that jointly accounts for detection and segmentation quality. Together, these metrics reflect the community's effort to capture not only pixel-level accuracy but also the structural and perceptual fidelity of predicted masks [67]. Despite their widespread use, these metrics also present important limitations. Pixel accuracy, for example, can be misleading in highly imbalanced datasets: if one class dominates (e.g., background in medical scans or "road" in driving scenes), a model may achieve high accuracy while essentially ignoring minority classes. Mean IoU (mIoU) alleviates this by averaging across categories, yet it still treats all classes equally, which may undervalue performance on rare but semantically crucial classes. The Dice coefficient, while well-suited for imbalanced medical tasks, is sensitive to small variations in boundary alignment, which may exaggerate performance differences. In instance segmentation, AP focuses on overlap quality but does not capture mask sharpness or boundary accuracy, and its heavy dependence on IoU thresholds can make comparisons across works inconsistent. Similarly, Panoptic Quality (PQ) offers a unified measure for panoptic segmentation, but it weights detection and segmentation errors in

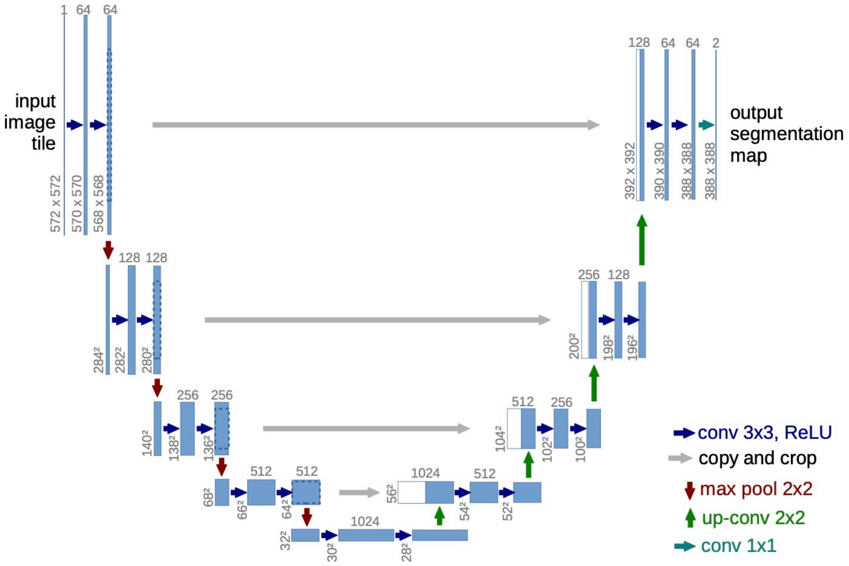


Figure 2.7: Figure from [6]. Architecture of the U-Net model. Blue box corresponds to a multi-channel feature map, while white one represents copied feature maps.

a fixed way; this can obscure cases where a method excels at one but not the other. As a result, while these metrics provide a common ground for benchmarking, they are not always sufficient to capture the nuanced trade-offs between accuracy, boundary fidelity, computational cost, and perceptual quality [67].

2.4.3 U-Net architecture

U-Net is one of the most influential architectures for semantic segmentation, first introduced by Ronneberger, Fischer, and Brox in the context of biomedical image analysis [6]. The model adopts a fully convolutional encoder–decoder design: the contracting path (encoder) extracts contextual and semantic features through successive convolution and pooling operations, while the expansive path (decoder) gradually upsamples the feature maps to recover spatial resolution. A key innovation of U-Net lies in the skip connections, which link corresponding layers of the encoder and decoder. These connections transfer fine-grained localization cues lost during downsampling and fuse them with the abstract semantic information learned at deeper layers, enabling accurate pixel-wise predictions. Figure 2.7 displays the architecture of the U-Net proposed by [6]. The blue boxes represent multi-channel feature maps, while white boxes correspond to copied feature maps. The arows in the bottom represent the several different operations applied on the feature maps.

The original U-Net was shown to outperform existing methods in the ISBI cell tracking challenge, even when only a small number of training samples was available [6]. This success was partly due to extensive data augmentation strategies, such as random elastic

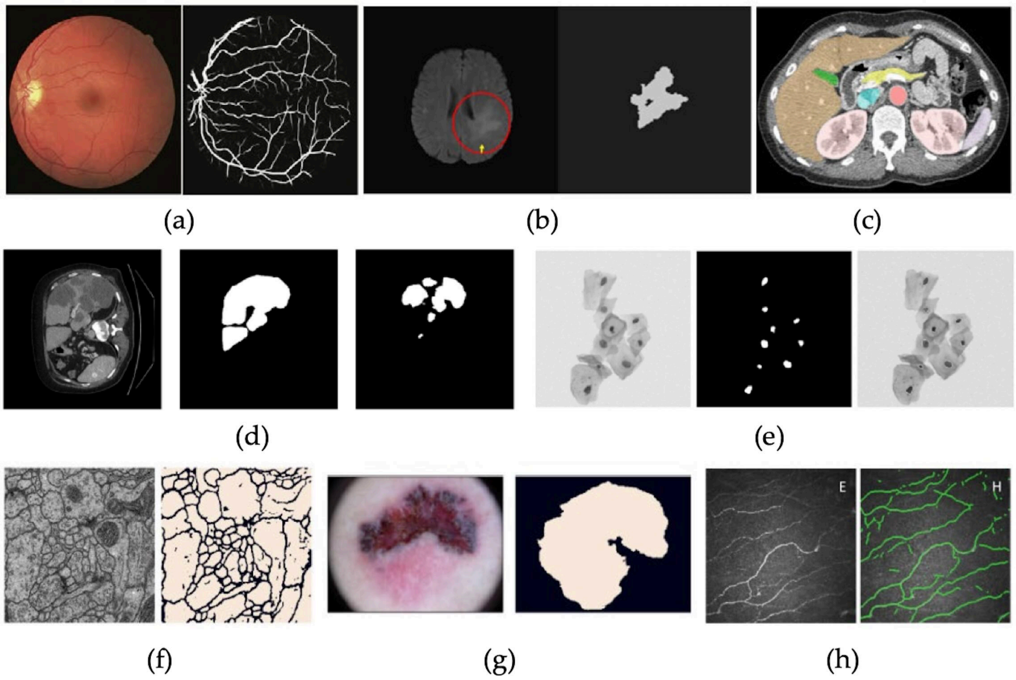


Figure 2.8: Figure from [95]. Examples of several U-Net application in medical field. (a) from [96]; (b) from [97]; (c) from [98]; (d) from [99]; (e) from [100]; (f) and (g) from [101]; (h) from [102].

deformations, which improved robustness to variability in biomedical images. Beyond the original 2D design, U-Net has been extended to handle volumetric data: Çiçek et al. [79] proposed the 3D U-Net, replacing all 2D operations with 3D convolutions to better capture spatial information across image slices, especially for Computed Tomography (CT) [91] and Magnetic Resonance Imaging (MRI) modalities [92, 93].

Further improvements targeted the skip connections themselves. Zhou et al. [78] introduced UNet++ , a nested architecture with redesigned skip pathways that allow multi-scale feature aggregation and more flexible fusion between encoder and decoder features. Oktay et al. [94] proposed the Attention U-Net, where attention gates are integrated into the skip connections to suppress irrelevant regions and highlight salient structures, a particularly useful property for medical images with high inter-patient variability . These works demonstrate the adaptability of the U-Net framework to different tasks and imaging modalities. Figure 2.8 displays several applications of the U-Net in medical and biological field.

Thanks to its modular and symmetric design, U-Net has also been applied far beyond biomedical segmentation, becoming a reference architecture in domains such as remote sensing, microscopy, and natural image processing. Variants now incorporate residual or dense backbones, multi-scale feature extraction, and even Transformer modules, but the core principle remains the same: a balanced encoder–decoder with skip connections to combine semantic context and precise localization. This combination has made U-Net and its derivatives the backbone of many modern segmentation pipelines.

Part II

Problem Statement and Experimental Framework

Chapter 3

Problem Statement and Dataset

In this chapter we define the problem addressed in this thesis and describe the dataset used in our work. In Section 3.1, we explain why developing an automated method for seabed mapping is important and outline what makes this study different from previous approaches. Our goal is to detect thin and elongated morphological elements (scarps, channels, etc.) and we highlight why this task is more challenging than standard image segmentation problems that usually deal with larger, compact shapes. Then, in Section 3.2 we describe the study area in the Ionian Sea (Figure 3.1) and report its geographic bounds (Table 3.2). We introduce the three input variables—depth, slope, and curvature—and show their maps (Figure 3.2) and value distributions (Figures 3.3 to 3.5). We present the labeling scheme derived from the MaGIC project, including the grouped classes and their detailed components (Table 3.1), and the corresponding label maps used as ground truth (Figure 3.6). Finally, we illustrate the simple thickening step applied to make lineaments more visible for learning (Figure 3.7), and we quantify class imbalance with the overall label frequencies (Figure 3.8). This material sets the stage for the model design, training protocol, and evaluation described in the following chapters.

3.1 Problem Definition

As introduced in Section 2.2, seabed morphology consists of a complex arrangement of features and elements that together record geological and geomorphological processes. The accurate interpretation of these elements has traditionally relied on expert judgment based on bathymetric and derivative maps. While this approach ensures geological coherence, it is inherently time-consuming and influenced by the interpreter's experience.

The growing availability of high-resolution bathymetric data now challenges the limits of manual mapping. Large datasets, such as those produced by the MaGIC Project [3], require new approaches capable of ensuring both efficiency and consistency across vast areas. This motivates the adoption of ML techniques as a complementary tool to tradi-

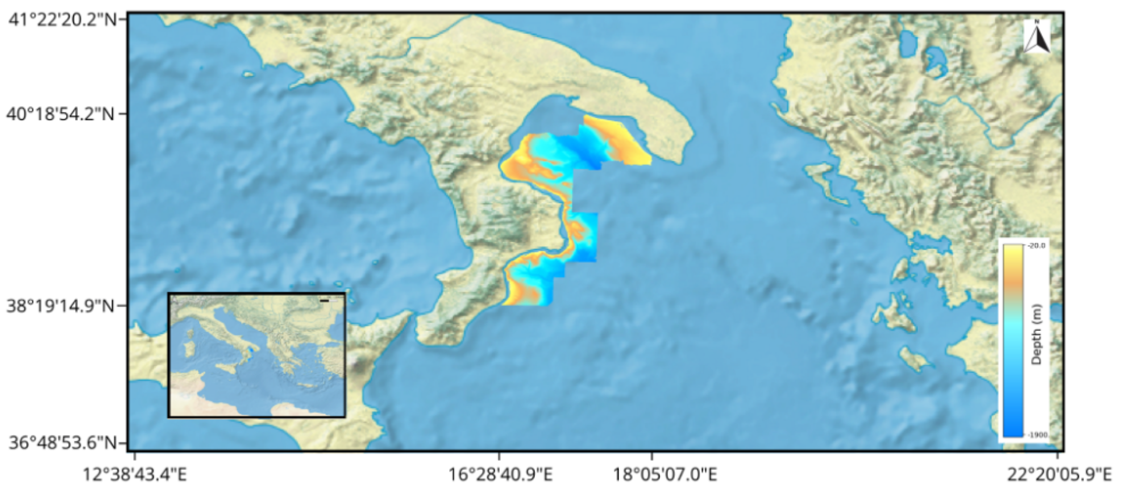


Figure 3.1: Digital Elevation model (DTM) of the Ionian Sea employed in this study, with latitude and longitude coordinates indicating the geographical coverage. The area under consideration is located in the Ionian Sea, along southern coasts of Italy.

tional interpretation.

However, adapting image segmentation methods to seabed classification presents specific challenges. Unlike standard segmentation tasks, often focused on compact, “blob-like” structures, seabed morphological elements are typically expressed as thin, elongated, and discontinuous lineaments. These kind of elements, together with the strong imbalance between background and target classes and the uncertainty inherent in expert annotations, make mapping the challenge very difficult.

The main objective of this thesis is therefore to develop and evaluate an automated ML framework for the classification of seabed morphological elements from bathymetric data. The proposed approach aims to support geologists in their interpretation tasks while providing a consistent and reproducible tool for marine geohazard assessment and infrastructure planning.

The following chapters describe how these challenges are addressed through specific design choices in data preprocessing, model architecture, and performance evaluation, with the ultimate goal of producing a geologically sound and operationally effective mapping framework.

3.2 Dataset Description and Analysis

This section describes the principal characteristics of the data used to train our model. The area of interest is the seabed along the southern coast of Italy, specifically between Calabria and Apulia, in the Ionian Sea, as shown in Figure 3.1.

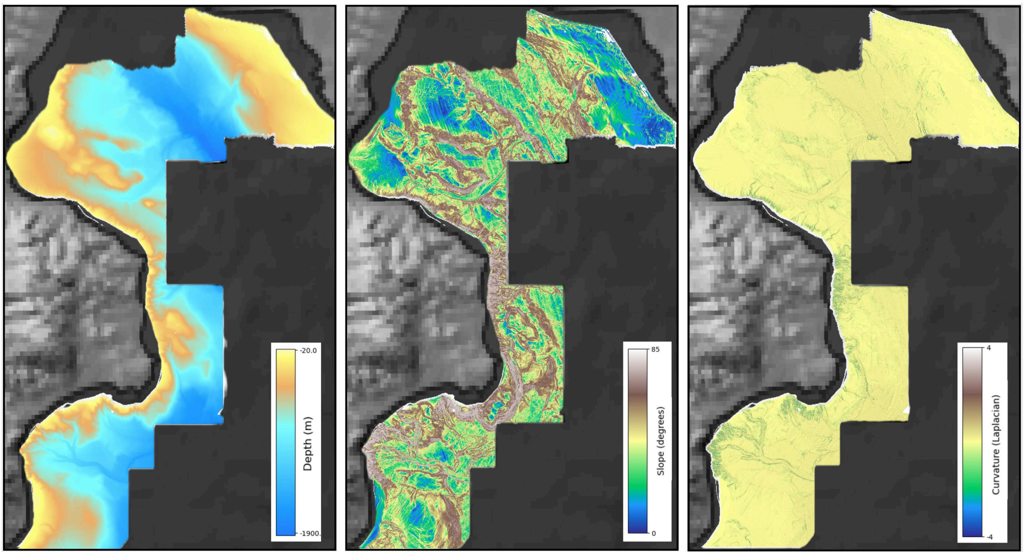


Figure 3.2: Depth, slope, and curvature maps of the study area extracted from high-resolution multibeam data, shown from left to right. All three maps refer to the same geographical region and represent different morphological features derived from the bathymetry.

Table 3.2: Minimum and maximum latitude and longitude values defining the geographical extent of the study area from which the data were collected.

	Min	Max
Latitude	38°19'14.9''N	40°18'54.2''N
Longitude	16°28'40.9''E	18°05'07.0''E

The exact coordinates of the area of interest are reported in Table 3.2, which lists the minimum and maximum values of latitude and longitude. The data used are primarily Geographic Information System (GIS) datasets–spatial data that represent real-world features along with their geographic locations. These datasets typically include coordinates, geometrical shapes (e.g., points, lines, and polygons), and associated attributes like population, elevation, or land use. GIS data are widely used in mapping, spatial analysis, and decision-making across various domains, including urban planning [103], environmental monitoring [104], and cancer mapping [105].

In our case, the dataset provides bathymetric information for the shallow coastal regions of Italy, i.e., the depth of the seabed [106, 107, 108], as visualized in Figure 3.1. Specifically, the data come from the Italian MaGIC project [3], a national initiative aimed at assessing seabed hazards in Italian waters to support risk mitigation and emergency management.

Starting from depth data, two additional features considered useful by domain ex-

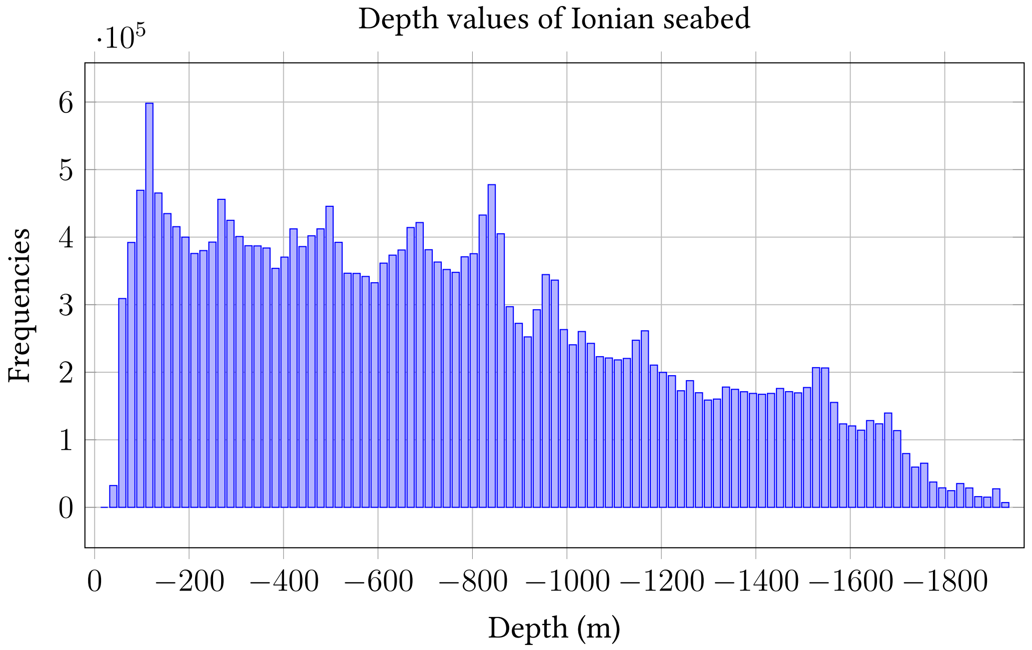


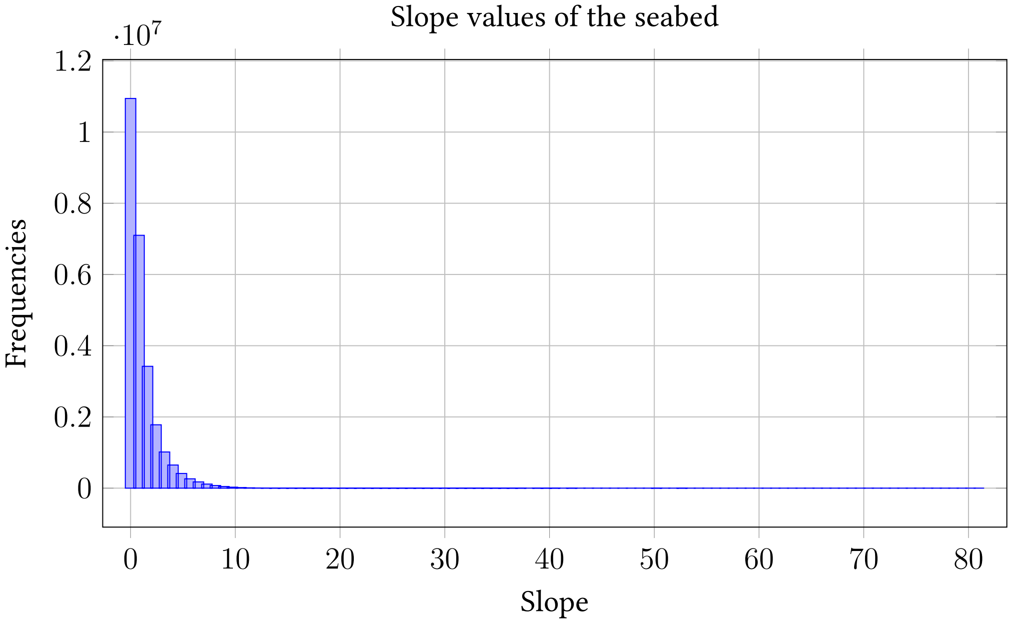
Figure 3.3: Bar plots showing the distribution of the depth values of the seabed of the area of interest. Data density increases as the depth becomes shallower.

perts can be derived: the *slope* and the *profile curvature* of the seabed, corresponding to the first and the second derivatives of depth, respectively, as shown in Figure 3.2. The resulting data can be interpreted a $2D$ map with three features–or channels–each representing a specific property of the same seabed area, making it directly suitable for image-based processing. As illustrated in Figure 3.2, a significant portion of the images contains grey pixels which indicate areas with missing data. This may occur for several reasons: some regions correspond to land rather than seafloor, data acquisition is not possible near the coastline due to shallow depth, and some areas were simply not covered during the survey. These feature maps are used as input to our model, as they capture essential information about the seafloor topography, which strongly influences the spatial distribution of morphological features.

Figure 3.3, 3.4 and 3.5 show the distribution of the values of depth, slope, and curvature for the dataset used in this study. As we can see from Figure 3.3 depth values range from about -20 to -1920 meters. The bar plot further highlights that the seafloor is not uniformly distributed across depths: very deep areas are relatively scarce, whereas shallow regions are more abundant, resulting in an overall increasing trend in the histogram. Figure 3.4(a) reports the raw slope values derived from the depth gradient. Most of the pixels have values close to zero, meaning that large portions of the seabed are relatively flat. To make the interpretation more intuitive, slope values were converted from gradient to degrees using the following transformation:

$$\text{slope}_{\text{deg}} = \arctan(\text{slope})\frac{180}{\pi} \tag{3.1}$$

(a) Distribution of the slope values



(b) Distribution of the slope values in degree

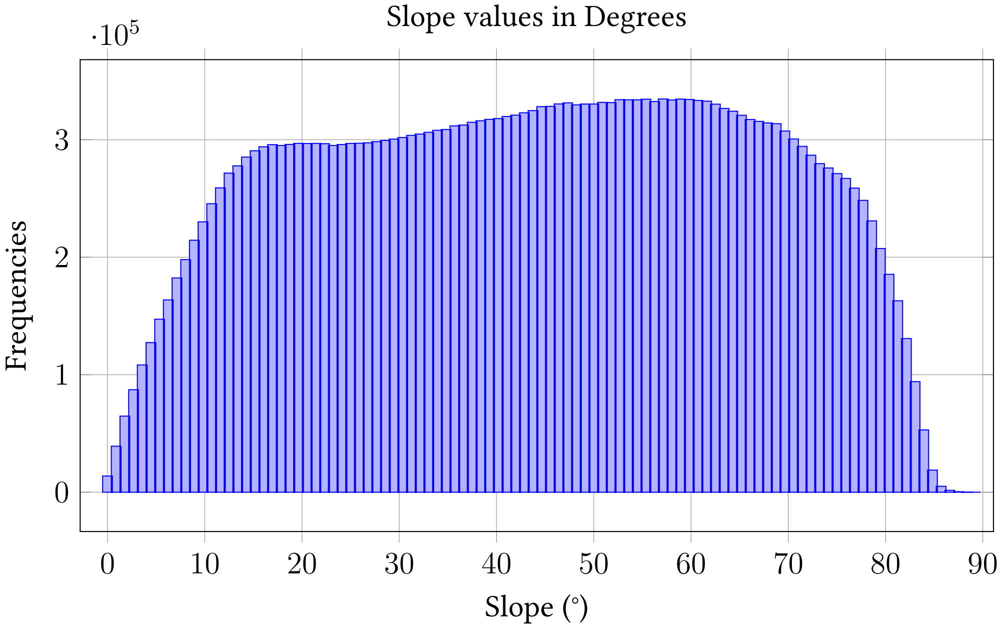


Figure 3.4: Bar plots showing the distribution of the slope values of the seabed of the area of interest. In (a) the slope is computed as the derivative of the depth. In (b) these data are transformed in degree, that are more uniformly distributed.

The resulting distribution, shown in Figure 3.4b, spans from 0 to 90 degrees. After this conversion, the slope values appear more evenly spread, with a clearer distinction between flat and steep areas.

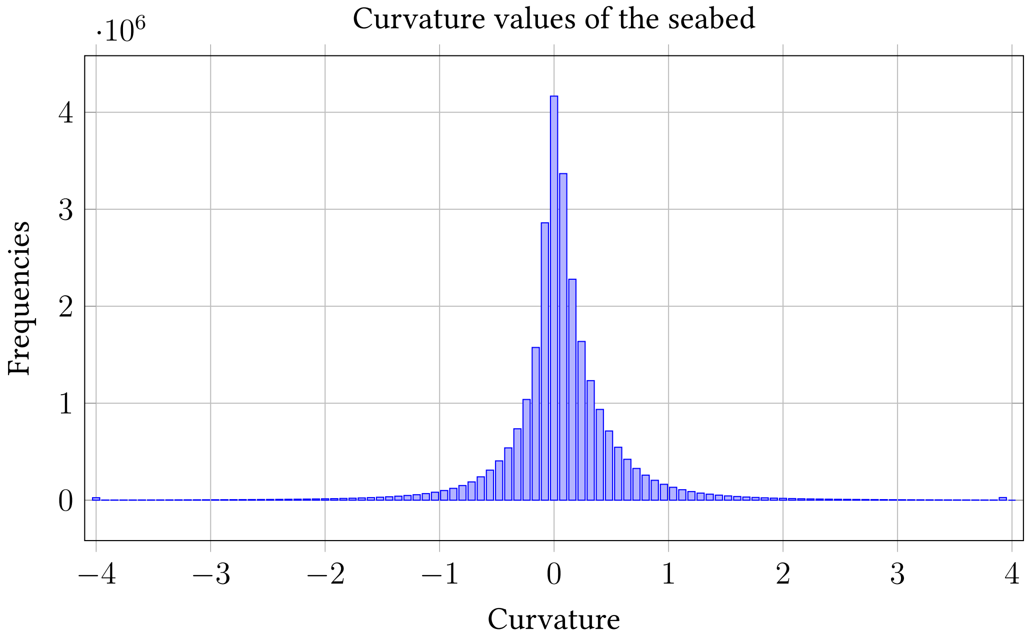


Figure 3.5: Bar plots showing the distribution of the curvature values of the seabed of the area of interest. The data distribution is highly peaked in zero and decreases very rapidly in both directions.

Finally, Figure 3.5 shows the distribution of curvature values. They are centered around zero and display an almost symmetric shape, with most values concentrated near zero. This means that flat or gently curved surfaces dominate the area, while strongly concave or convex features are less frequent.

The data are organized as an image of 11000×6000 pixels, as shown on the left in Figure 3.6. For clarity and ease of interpretation, only the seabed depth will be displayed from this point onward, while the slope and profile curvature—although used during training—will be omitted from the visualizations. Land areas have also been removed to better highlight the recorded data (i.e., the colored pixels).

All the data produced by the MaGIC project [3] also received human interpretations of the seabed morphology i.e., a *labels map* of the same dimensions of the input 2D fields indicating the location of all morphological elements in that region, as shown in the right panel of Figure 3.6. Each label corresponds to a specific seabed feature and is represented in the map as a line. This annotation is currently performed manually by geologists, who analyze the bathymetry and delineate the corresponding features directly using GIS software, identifying and tracing the boundaries of distinct morphological structures. In the original dataset, each type of line corresponds to one of 49 classes, as defined by the marine scientists involved in the MaGIC project [3]. However, due to the large number of classes and the limited representation of many of them, a first pre-processing step was applied to reduce the number of classes to 11. These 11 groups displayed in Figure 3.6, were obtained by clustering the original 49 classes based

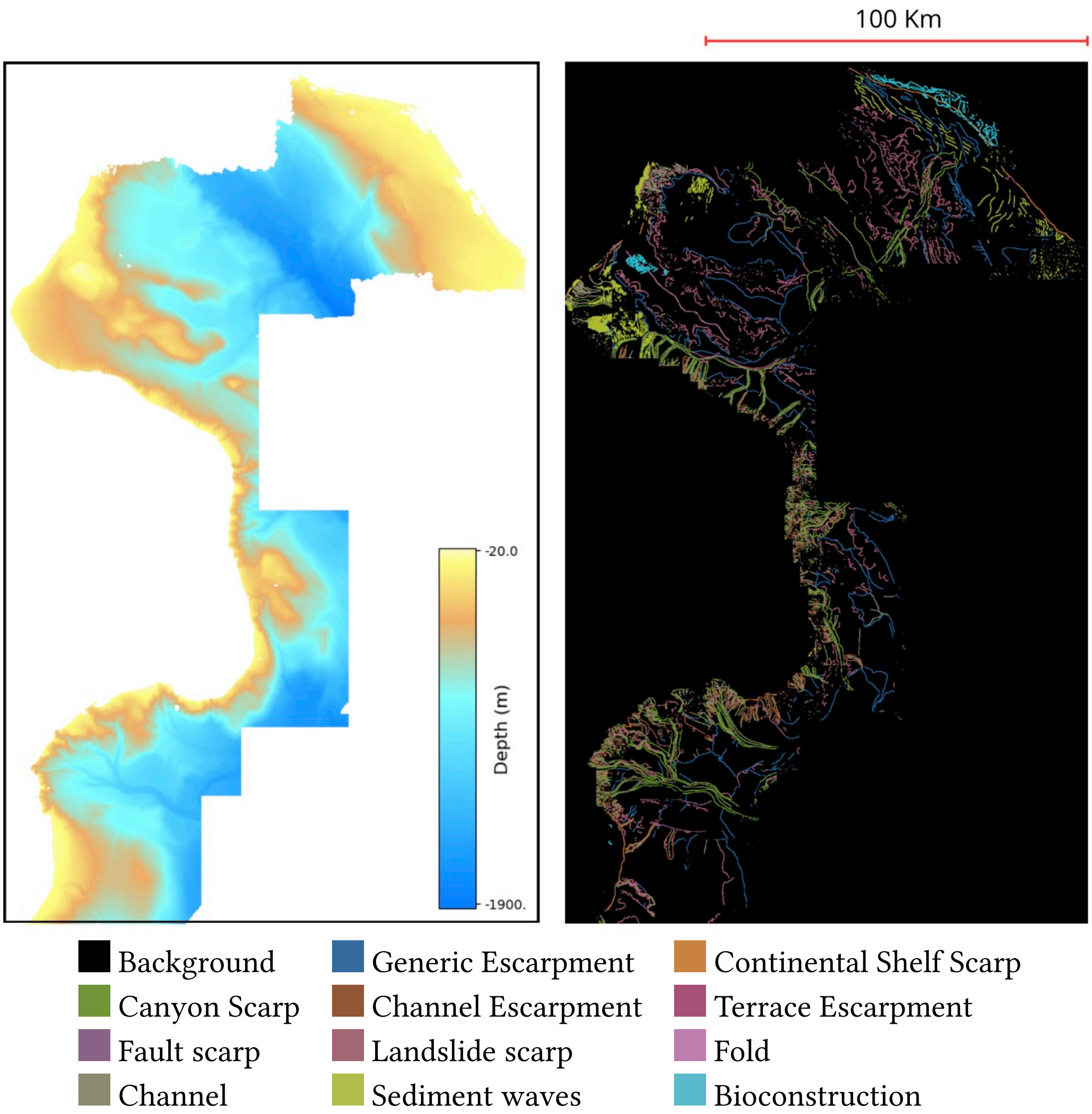
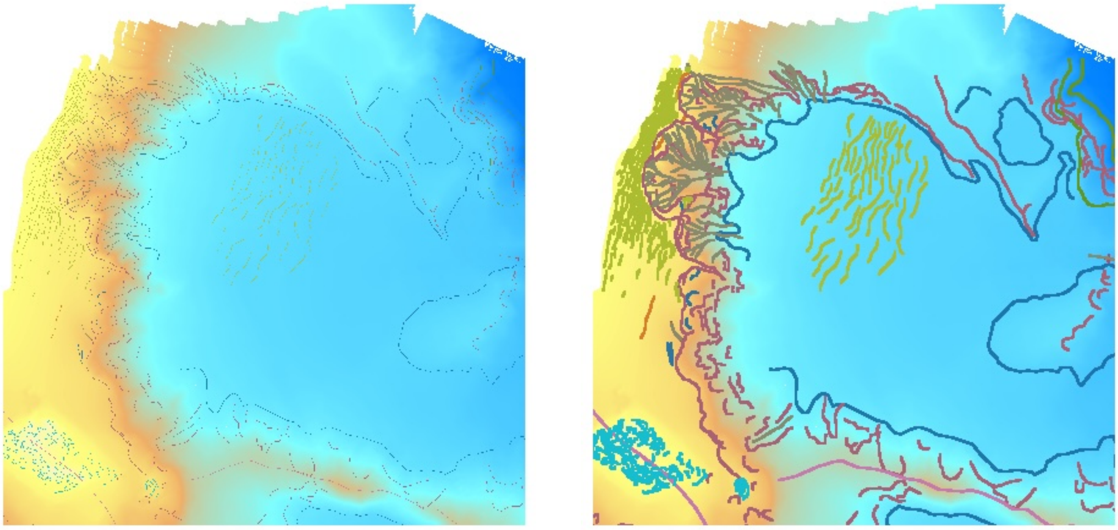


Figure 3.6: The figure shows the input data used in this study. On the left is the bathymetric map of the area, while on the right is the geological interpretation provided by experts, illustrating the main morphological elements identified. The different elements are listed in the legend. Background corresponds to no element in that specific point.

on geological and morphological similarity, with guidance from domain experts. Since all annotations are line-based, the majority of pixels do not belong to any specific class and are labeled as background. Consequently, the output of our model is a 2D field in which each pixel is assigned to one of 12 classes—11 corresponding to morphological elements, and 1 representing the background. We emphasize that background pixels indicate the absence of any annotated morphological features and thus represent regions of no particular geological interest. These *labels map* serve as the ground truth during training, meaning that they represent the desired output that the model is expected to learn to reproduce. Given a features map as input, the model is trained to generate a corresponding labels map that indicates the morphological elements present in the area.

As mentioned above, our ground truth data consists of hand-drawn lines, each corresponding to one of the 11 classes. These lines are generally thin and sparse compared to the overall image area. To partially reduce this problem, we made the lineaments, that represent morphological elements, thicker through a simple process that adds pixels of the same class next to each non-zero pixel. This operation results in thicker and more continuous lines.



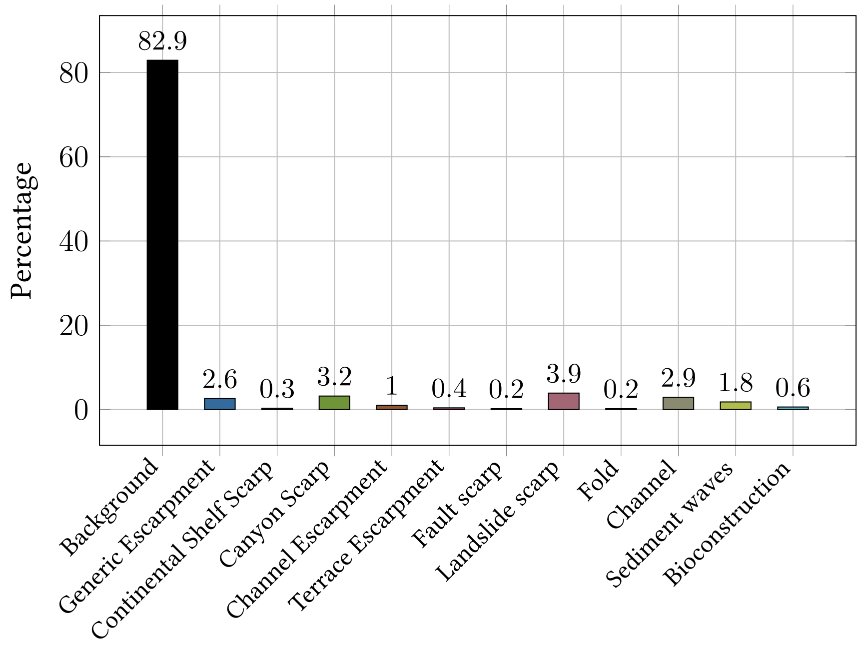
(a) Original lineaments (b) Lineaments after the thickening process

Figure 3.7: Comparison between the original and the processed lineaments on a small part of the MaGIC dataset. (a) Original lineaments, which appear thin, fragmented, and barely visible. (b) Lineaments after the thickening process, appearing more continuous and uniform.

The outcome of this process is shown in Figure 3.7, which represents a small portion of the total study aera. In 3.7a, the original lineaments are displayed: they are barely visible and highly fragmented. In contrast, Figure 3.7b shows the processed version, where the lineaments appear thicker and more uniform.

Despite this, the majority of the dataset is composed of background pixels, which do not belong to any of the 11 classes, resulting in most pixels being labeled as background (class 0). This leads to a highly imbalanced dataset dominated by background pixels. Figure 3.8 displays two bar plots: the first shows the frequency of all classes, including the background, while the second focuses only on the morphological classes. As shown, over 80% of the pixels belong to the background class (label 0), highlighting a significant class imbalance in the dataset. Some classes, such as Continental Shelf Scarp and Fault Scarp, account for less than 1% of the total. Even when excluding the background (right plot), class imbalance persists. In particular, Landslide Scarp and Canyon Scarp appear most frequently, with approximately 22.8% and 18.9% of the labeled pixels, respectively, while several other classes remain close to 1%.

(a) Distribution of the labels



(b) Distribution of the labels without background

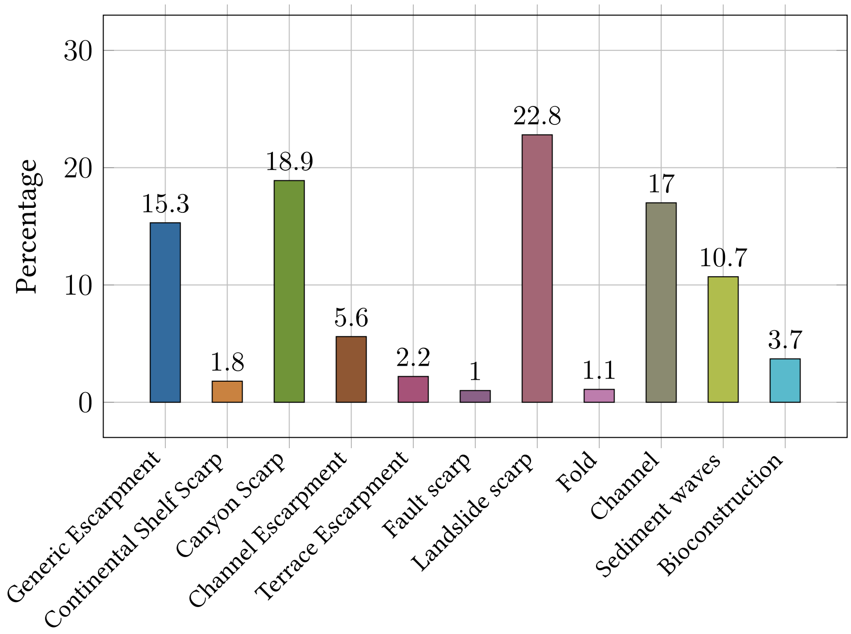


Figure 3.8: Bar plots showing the distribution of labels in the dataset. (a) plot includes the background class, while (b) plot excludes it. Dataset is highly unbalanced even when background is removed

Table 3.1: Correspondence between grouped morphological features and their detailed components.

Color	Grouped Feature	Detailed Features
🟦	Generic Escarpment	Undefined escarpment (Sharp) Undefined escarpment (Rounded) Undefined escarpment (Base) Generic erosion escarpment (Sharp) Generic erosion escarpment (Rounded) Generic erosion escarpment (Base)
🟧	Continental Shelf Scarp	Continental shelf escarpment (Sharp) Continental shelf escarpment (Rounded)
🟩	Canyon Scarp	Canyon/Channel escarpment (Sharp) Canyon/Channel escarpment (Rounded) Canyon/Channel escarpment (Base)
🟫	Channel Escarpment	Minor channel escarpment (Sharp) Minor channel escarpment (Rounded) Minor channel escarpment (Base)
🟥	Terrace Escarpment	Intrachannel terrace escarpment (Edge) Intrachannel terrace escarpment (Base) Intrachannel step escarpment (Edge) Intrachannel step escarpment (Base)
🟪	Fault scarp	Fault escarpment (Rounded) Fault escarpment (Base)
🟥	Landslide scarp	Simple landslide niche escarpment (Edge) Simple landslide niche escarpment (Base) Complex landslide niche escarpment (Edge) Complex landslide niche escarpment (Base) Intrachannel landslide niche escarpment (Edge)
🟪	Fold	Compression fold ridge (Axis)
⬛	Channel	Concave-bottomed channel V-shaped channel Erosive gully
🟨	Sediment waves	Sediment waves (Axis)
🟦	Bioconstruction	Bioconstruction

Chapter 4

Methodology

Introduction

This chapter describes the methodology adopted in this work, including the model architecture, data preparation, training procedure, and evaluation strategy. The main goal is to adapt a CNN for the automatic classification of seabed morphological elements, taking into account the specific challenges of this type of data: the thin and sparse nature of the features, the strong class imbalance, and the presence of interpretative uncertainty in the labels.

Section 4.1 introduces the general structure of the proposed framework, based on a U-Net architecture commonly used for image segmentation. The following sections 4.2 provides a overview of the workflow used for the experiments. It explain how the input data are divided into smaller patches, how overlapping windows are used to increase the number of samples, and how output data are post-processed after the training to improve the continuity and readability of the predictions. It also describes the training phase and the losses used in the experiments. The last part provides a description of the evaluation criteria, and try to motivate why pixel-based measures such as Dice coefficient and Cross Entropy, are not very appropriate for this task. A neighborhood-based bidirectional metric is introduce to better reflect the way experts compare predicted and real morphological features.

These elements define the complete workflow that will be used to train, validate, and test the models discussed in the following chapter.

4.1 U-net-based Architecture

We first describe the U-Net-based segmentation architecture along with the post-processing steps used to refine the output and improve label accuracy. We then outline the training

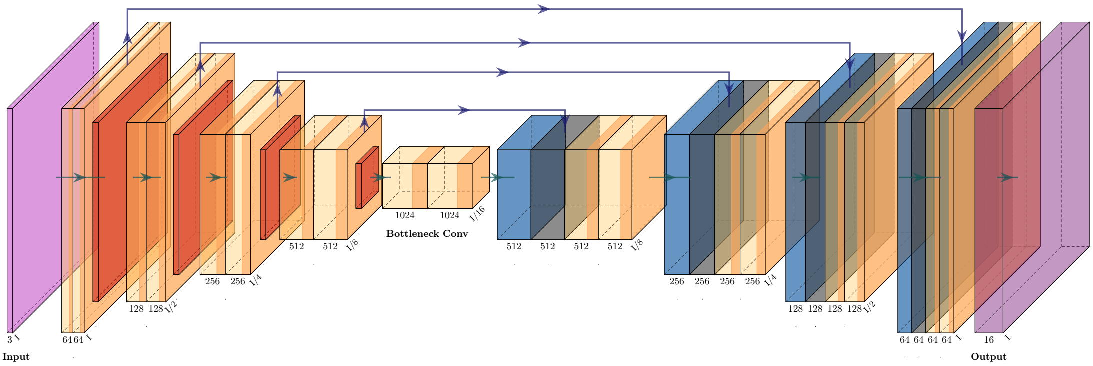


Figure 4.1: Network architecture used in this work. The yellow and blue boxes represent convolutional and deconvolutional layers, respectively. The *ReLU* functions are represented by the orange part of the boxes. The red squares correspond to *max pooling* layers.

procedure and the loss functions employed.

As mentioned above, we employ CNNs, a class of DL models specifically designed to process grid-structured data such as images [4]. CNNs extract spatial features through local receptive fields, weight sharing, and hierarchical processing via convolution and pooling layers. These properties make them both computationally efficient and robust to common image transformations such as translation, rotation, and scaling [4] Originally introduced for image classification, CNNs have become the foundation of most modern computer vision systems, including tasks as object detection, face recognition and image segmentation [4].

Our focus is on semantic segmentation, where the goal is to assign a class label to each pixel of the input image. For this task, we adopt a U-Net based architecture [109], a widely used model in biomedical and environmental imaging. U-Net is characterized by a symmetric encoder-decoder structure and skip connections, which allow the decoder to recover spatial precision by combining low-resolution semantic information with high-resolution features from the encoder. As shown in Figure 4.1, the architecture consists of three main components: (i) a contracting path (encoder), that reduces the spatial resolution while increasing the number of feature maps; (ii) an expansive path (decoder), which upsamples the feature maps to recover the original resolution; (iii) skip connections, which link encoder and decoder layers at corresponding resolutions to preserve spatial context.

The specific parameters of the different layers are summarized in Table 4.1. The input tensor has shape $[3, 100, 100]$, corresponding to RGB images of size 100×100 pixels. The encoder path consists of five stages. Each stage applies two successive convolutional operations with a kernel size of 3×3, stride 1, and padding 1. Each convolution is followed by batch normalization, which improves training stability and convergence speed by mitigating internal covariate shift [110], and by the *ReLU* activation function [59]:

$$\mathrm{ReLU}(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{4.1}$$

In the contracting path, each stage ends with a *max pooling* operation with a 2×2 kernel, except for the final one. This pooling step halves the spatial resolution, reducing redundant information while retaining the most relevant features [4]. The output is then processed by two convolutional layers with the same kernel size, padding, and activation structure as in the first stage.

The decoder mirrors the encoder with four upsampling blocks. Each begins with a transposed convolution (deconvolution) that doubles the spatial resolution. The output is concatenated with the corresponding feature maps from the encoder via skip connections. The concatenated tensor is processed by two convolutional layers, again with batch normalization and ReLU activation.

The final layer is a 1×1 convolution that projects the feature maps to the number of output classes. In our case, this results in a tensor of shape $[12, 100, 100]$, corresponding to 12 seabed element classes. During inference, the output logits are converted to class probabilities using the softmax function.

4.2 Pipeline flowchart

The training pipeline consists of multiple stages. First, in Section 4.2.1, we describe the preprocessing steps used to generate the data fed into the network from the maps shown in Figure 3.6. In Section 4.2.2, we then outline the training phase, detailing the loss functions, optimization strategy, and relevant hyperparameters. Section 4.2.4 presents the qualitative and quantitative methods used to evaluate model's performance.

4.2.1 Input preprocessing

As described in Section 3.2, the input consists of a three-channel image covering the study area (depth, slope, and curvature; see Figure 3.2). First, each channel is normalized to have values between 0 and 1. The dataset is split $80\% - 20\%$ into training and test sets. To ensure that the training and test regions are sufficiently large and spatially separated, we first extract random 500×500 pixel regions from the full feature maps, with 20% of these regions set aside for testing. Because using 500×500 images directly would result in too few training (and testing) samples, each large region is further divided into smaller 100×100 patches, which are the actual input samples for the network. Specifically, within each 500×500 training region, a sliding window of size 100 × 100 with a stride of 50 pixels is applied. As previously mentioned, the bathymetry data have a spatial resolution of 20 meters. Consequently, each input patch spans an area of 4 km^2,

Table 4.1: Layer-wise specification of the model architecture used in this study. Each layer is annotated with kernel size (k), stride (s), padding (p), output dimensions, and operations. The encoder spans layers 1–9, the decoder layers 10–21, and layer 22 is the final projection.

Layer	Type	k	s	p	Output Size	Details
Input	-	-	-	-	[3, 100, 100]	Input image
1	Conv Block (×2)	3×3	1	1	[64, 100, 100]	2×Conv + BN + ReLU
2	Max Pool	2×2	2	0	[64, 50, 50]	Downsample
3	Conv Block (×2)	3×3	1	1	[128, 50, 50]	2×Conv + BN + ReLU
4	Max Pool	2×2	2	0	[128, 25, 25]	Downsample
5	Conv Block (×2)	3×3	1	1	[256, 25, 25]	2×Conv + BN + ReLU
6	Max Pool	2×2	2	0	[256, 12, 12]	Downsample
7	Conv Block (×2)	3×3	1	1	[512, 12, 12]	2×Conv + BN + ReLU
8	Max Pool	2×2	2	0	[512, 6, 6]	Downsample
9	Conv Block (×2)	3×3	1	1	[1024, 6, 6]	2×Conv + BN + ReLU
10	Transposed Conv	2×2	2	-	[512, 12, 12]	Upsample
11	Skip Connection	-	-	-	[1024, 12, 12]	From layer 7
12	Conv Block (×2)	3×3	1	1	[512, 12, 12]	2×Conv + BN + ReLU
13	Transposed Conv	2×2	2	-	[256, 24, 24]	Upsample
14	Skip Connection	-	-	-	[512, 25, 25]	From layer 5 (aligned)
15	Conv Block (×2)	3×3	1	1	[256, 25, 25]	2×Conv + BN + ReLU
16	Transposed Conv	2×2	2	-	[128, 50, 50]	Upsample
17	Skip Connection	-	-	-	[256, 50, 50]	From layer 3
18	Conv Block (×2)	3×3	1	1	[128, 50, 50]	2×Conv + BN + ReLU
19	Transposed Conv	2×2	2	-	[64, 100, 100]	Upsample
20	Skip Connection	-	-	-	[128, 100, 100]	From layer 1
21	Conv Block (×2)	3×3	1	1	[64, 100, 100]	2×Conv + BN + ReLU
22	Final Conv	1×1	1	0	[12, 100, 100]	Output logits

BN: Batch Normalization. ReLU: Rectified Linear Unit. Conv: Convolution.

which represents a balanced compromise between computational efficiency (in terms of memory and training time) and the spatial coverage needed to capture and recognize the typically large-scale morphology of geological features.

Each patch therefore has shape $[3, 100, 100]$. This approach ensures that test areas remain distinct and reasonably large. Large regions were first selected and then subdivided into smaller patches, ensuring that the training and test sets remained completely separate, even when overlapping windows were used during training. Within each 500×500 training region, a sliding window of size 100×100 was applied with a stride of 50 pixels. This procedure generated multiple partially overlapping patches per region, thereby increasing the diversity of the training data while still guaranteeing that no patch overlapped with the designated test areas. Importantly, patches were retained only if at least 70% of their pixels contained valid data (i.e., not white background).

To assess the role of overlap, simulations were performed both with and without overlapping windows. As will be shown later, introducing overlap leads to improved results, since the model benefits from a denser sampling of local structures and from increased variability in the training set.

Each input patch is paired with a corresponding label patch, which is extracted by applying the same sliding window procedure on the corresponding 500×500 label region. Label patches have shape $[12, 100, 100]$, where the 12 channels represent the 11 morphological classes plus background. Overall, this procedure yields about 6000 training samples.

Furthermore, models with simplified classification schemes were also trained. In particular, one model was trained with a reduced grouping of features compared to that reported in Table 3.1, where all escarpments and scarps were merged into a single class as shown in Table 4.2. In addition, another model was trained to perform a binary classification, distinguishing only between background and lineaments, with all lineaments assigned the same label. In those specific cases the label patches have shape $[6, 100, 100]$ and $[2, 100, 100]$.

In the following sections, the general training and post-processing procedure is described, which remains the same across all cases.

4.2.2 Training

The model is trained on 100×100 patches with three input channels and returns a $[12, 100, 100]$ output tensor. For each pixel, the network produces 12 logits, one for each class. The final class is determined by selecting the channel with the highest score.

The loss function combines standard cross-entropy with the Dice loss [111, 112], which measures the overlap between predicted and target masks. The overall loss func-

Table 4.2: Simplified grouped morphological features and their detailed components. This is a simplified version of the grouping reported in Table 3.1, where *escarpments* and *scarps* are merged togheter. In this case 6 different morphological classes are obtained.

Color	**Grouped Feature**	**Detailed Features**
🟦	Generic Escarpment	Undefined escarpment (Sharp) Undefined escarpment (Rounded) Undefined escarpment (Base) Generic erosion escarpment (Sharp) Generic erosion escarpment (Rounded) Generic erosion escarpment (Base) Continental shelf escarpment (Sharp) Continental shelf escarpment (Rounded) Canyon/Channel escarpment (Sharp) Canyon/Channel escarpment (Rounded) Canyon/Channel escarpment (Base) Minor channel escarpment (Sharp) Minor channel escarpment (Rounded) Minor channel escarpment (Base) Intrachannel terrace escarpment (Edge) Intrachannel terrace escarpment (Base) Intrachannel step escarpment (Edge) Intrachannel step escarpment (Base) Fault escarpment (Rounded) Fault escarpment (Base) Simple landslide niche escarpment (Edge) Simple landslide niche escarpment (Base) Complex landslide niche escarpment (Edge) Complex landslide niche escarpment (Base) Intrachannel landslide niche escarpment (Edge)
🟧	Fold	Compression fold ridge (Axis)
🟩	Channel	Concave-bottomed channel V-shaped channel Erosive gully
🟫	Sediment waves	Sediment waves (Axis)
🟪	Bioconstruction	Bioconstruction

tion is given by:

$$L_{\text{tot}} = L_{\text{CE}} + L_{\text{dice}} \tag{4.2}$$

The Dice coefficient c_{dice} is defined as:

$$c_{\text{dice}} = 1 - L_{\text{dice}}. \tag{4.3}$$

This coefficient ranges from 0 to 1 and provides an interpretable measure of segmentation quality.

Training is performed for 2000 epochs using the Adam optimizer, with a batch size of 128. The learning rate starts at 0.1 and decays to 1×10^{-5} following a cosine annealing schedule [113]:

$$\text{lr}(t) = \eta_{\text{min}} + \frac{1}{2}(\eta_{\text{max}} - \eta_{\text{min}}) \left(1 + \cos\left(\frac{t}{T_{\text{max}}}\pi\right)\right) \tag{4.4}$$

where $T_{\text{max}} = 1500$ and $\eta_{\text{min}} = 10^{-5}$. This setup allowed the model to gradually refine its learning while maintaining stability during the later stages of training.

4.2.3 Post processing

In some cases, the model tends to predict small, isolated blobs that clearly do not correspond to any real physical morphological structures. To address this issue, we applied a simple *cleaning algorithm* to remove such artifacts. Specifically, the algorithm identifies and removes all connected regions (blobs) composed of fewer than 400 pixels. The cleaning process works by considering all connected non-background regions less than 400 pixels (corresponding to an area of 0.16 km^2) in the predicted map and assigning them as background. This helps improve the realism and physical consistency of the final output.

4.2.4 Evaluation

Standard metrics such as the Dice coefficient and Cross-Entropy loss do not fully capture the quality of the results in this context. One important aspect to notice is that, for the proposed results, the Cross Entropy loss is not a good indicator of the usefulness of the results. In fact, it represents only a proxy of the real usefulness of the proposed labeling, since labeling by experts is itself a subjective and noisy act. Thus, obtaining a feature that is shifted by a small amount might be immaterial (the original lines were themselves drawn by hand), like getting a non-continuous line (the expert interpretation would be that there should be a “connection” between two features). Hence, most of the evaluation is still qualitative and based on discussions with experts, who are particularly interested

Algorithm 1: Bidirectional neighborhood-based complementary metric for a single image pair

```
Input: Ground truth image $G$, reconstructed image $R$, neighborhood radius $\epsilon$
Output: $\mathrm{Score}_{G\to R}$, $\mathrm{Score}_{R\to G}$, label-wise percentages

Function ComputeSuccessAndLabels(Source $S$, Target $T$):
    $P \leftarrow \{(x,y) \mid S(x,y) > 0\}$ ;          // Non-zero pixels in source
    foreach $(x,y) \in P$ do
        $\ell \leftarrow S(x,y)$ ;
        Increment count $C_\ell$ ;
        Define neighborhood $N_T(x,y) = T[x-\epsilon : x+\epsilon,\ y-\epsilon : y+\epsilon]$ ;
        if $\ell \in N_T(x,y)$ then
            Increment success $S_\ell$ and total success counter;
    foreach label $\ell$ do
        $\mathrm{Success}_\ell \leftarrow S_\ell / C_\ell$
    $\mathrm{Score}_{S\to T} \leftarrow$ total successes/$|P|$ return $\mathrm{Score}_{S\to T}$, $\{\mathrm{Success}_\ell\}$

Forward direction: $(\mathrm{Score}_{G\to R},\ \mathrm{ByLabel}_{G\to R}) \leftarrow$
 ComputeSuccessAndLabels($G, R$);
Backward direction: $(\mathrm{Score}_{R\to G},\ \mathrm{ByLabel}_{R\to G}) \leftarrow$
 ComputeSuccessAndLabels($R, G$);

return $\begin{cases} \textit{Total: } (\mathrm{Score}_{G\to R},\ \mathrm{Score}_{R\to G}) \\ \textit{By-label: } (\mathrm{ByLabel}_{G\to R},\ \mathrm{ByLabel}_{R\to G}) \end{cases}$
```

in the ability of the model to produce the correct "general shape" of the features, more than some specific details.

Another evaluation metric employed in this study is the confusion matrix, a widely used tool that summarizes the performance of a classification model by reporting the number of correct and incorrect predictions for each class [59]. However, this metric is inherently pixel-wise, and therefore it shares the same limitations discussed above: small spatial shifts in the predicted features, discontinuities, or minor misalignments with respect to the hand-drawn ground truth may result in a significant number of misclassified pixels, even though the overall shape of the feature is preserved. As a consequence, while the confusion matrix provides useful quantitative information, it does not fully capture the qualitative aspects that are often most relevant in expert-driven geological interpretation.

Since the ground truth consists of lines with minimal thickness, even a correct prediction shifted by a single pixel is penalized more than predicting only background. To better reflect the practical usefulness of the predictions, we introduce a complementary metric, denoted as c.

The proposed metric works as follows: given a predicted map, each point with a non-zero label is checked for the presence of the same label within a neighborhood of radius ϵ in the ground truth. The percentage of matched points is recorded. The process is then repeated in the opposite direction, starting from the ground truth and verifying matches in the prediction.

The exact algorithm for a single couple of images is described in Algorithm 1. When multiple images are considered, the coefficients are computed by summing the number of successful matches for each label across all images and dividing by the total number of pixels of that label in all images. This is equivalent to taking a weighted average of the coefficients from each image, where the weight is given by the number of pixels of that specific label present in each image.

This yields two complementary values ($c_{\text{gt-rec}}$ and $c_{\text{rec-gt}}$) that indicate the agreement between prediction and ground truth within a tolerance radius ϵ. By definition, the values of these two coefficients range from 0 to 1, where a value of 0 indicates poor performance and a value of 1 represents the maximum achievable performance. Higher values therefore correspond to better agreement between predictions and ground truth, reflecting a more accurate and reliable model. This metric can be viewed as a relaxed type of a pixel-wise accuracy, related only to the non-zero labels. The two directional components of the proposed metric, $c_{\text{gt-rec}}$ and $c_{\text{rec-gt}}$, are also conceptually analogous to recall and precision, respectively. In classical binary classification, recall quantifies how many of the true positive elements are correctly identified among all the actual positives, while precision measures how many of the predicted positives correspond to true ones. In our spatial formulation:

- $c_{\text{gt-rec}}$ acts as a geometric recall, expressing the fraction of the ground-truth pixels (non-zero labels) that are successfully detected within a tolerance radius ϵ in the predicted map.
- $c_{\text{rec-gt}}$ behaves as a geometric precision, quantifying the proportion of predicted pixels that correspond to real features in the ground truth.

Chapter 5

Experiments and Results

This chapter presents the experiments carried out to evaluate the proposed framework on different classification settings. Three main configurations are tested: a full multi-class model, a reduced-class model with a simplified grouping of features, and a binary model that distinguishes only between background and lineaments. Each configuration allows us to explore different aspects of the problem and to understand how class complexity influences model performance and interpretability.

Section 5.1 focuses on the main multi-class model, which includes all morphological elements described in the Table 3.1. Quantitative and qualitative results are presented, highlighting the model's ability to reproduce the main features identified by experts and the classes that remain more challenging.

Section 5.2 introduces the reduced-class and binary models. These simplified configurations aim to test whether merging similar classes or focusing only on lineament detection can lead to more stable and general results. For all configurations, model performance is analyzed using both standard pixel-based metrics and the bidirectional metric described in Chapter 4, as well as through comparisons with expert interpretations. The chapter includes an analysis of model behavior as a function of geomorphic variables such as depth, slope, and curvature, providing insight into the conditions under which the models perform best.

Finally, the results are critically discussed and contextualized. Section 5.3 examines the main findings of the experiments, comparing the three model configurations and highlighting how class aggregation influences prediction of the features and generalization. Section 5.4 outlines the main challenges encountered during training and testing, such as data imbalance, class ambiguity, and the difficulty of distinguishing morphologically similar features. Overall, this chapter aims to evaluate the effectiveness, robustness, and practical usefulness of the proposed approach across different levels of classification detail.

5.1 Main multi-class model

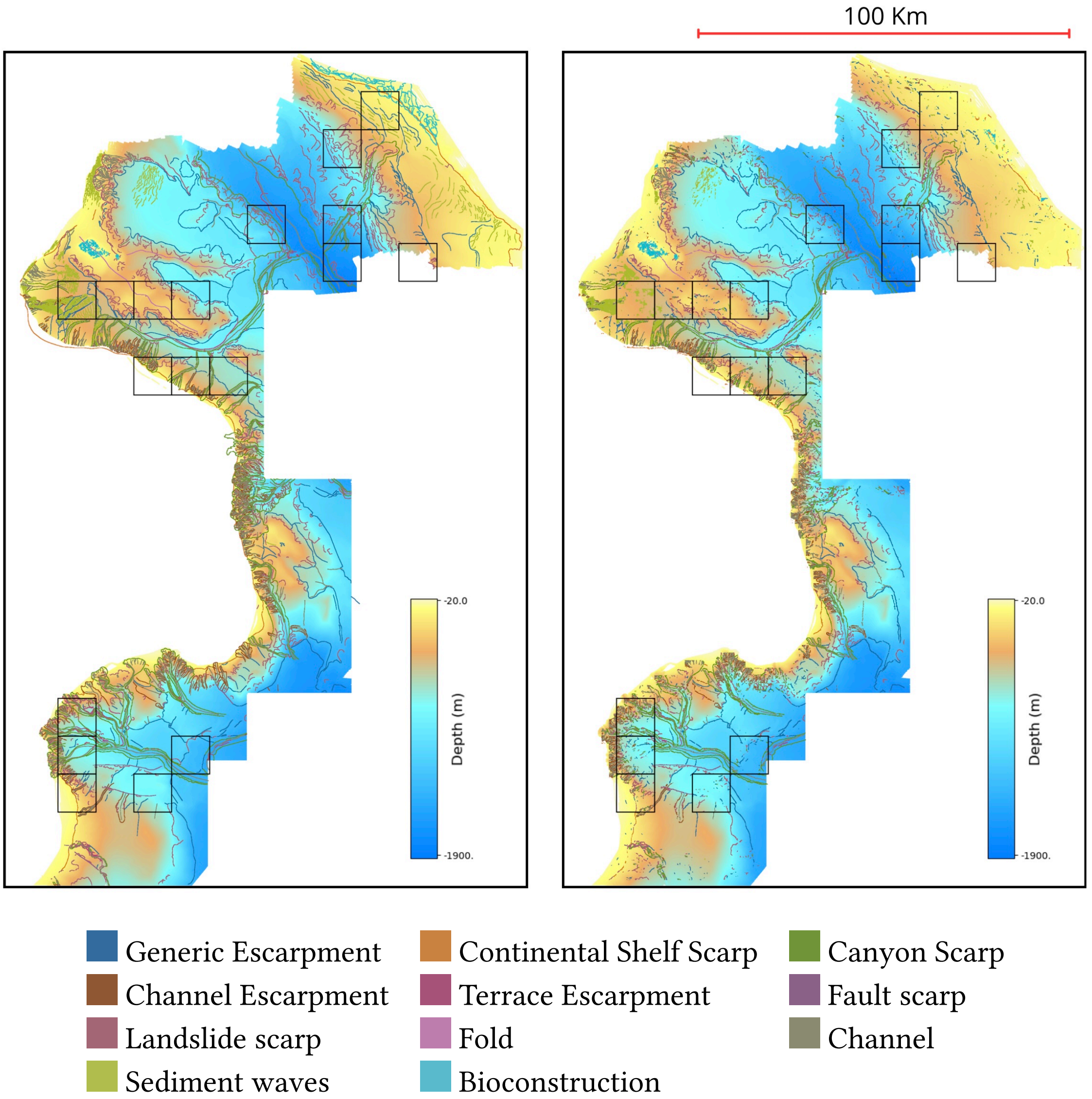


Figure 5.1: Human interpretation (left) and predicted map (right) for one region of the dataset. Black squares indicate test regions excluded from training phase.

In this section, we present the first results to assess whether the model can provide meaningful seabed classifications and support expert interpretation.

Figure 5.1 compares the human interpretation with the map reconstructed by the model. The black squares highlight areas held out during training, used for testing. This visual comparison illustrates the model's ability to reproduce the main morphological structures drawn up by experts in the domain. In general, the model performs better in the training areas compared to the test areas, as expected. However, even in the training set, the predicted linear features are often fragmented and not continuous, appearing as disconnected segments. Moreover, small isolated blobs, which do not correspond to any

Table 5.1: Comparison of class frequencies in the test areas. The first column reports the percentage of pixels belonging to each class in the ground truth maps, while the second column shows the corresponding frequencies in the reconstructed maps predicted by the model.

Color	Label	Ground Truth freq. (%)	Reconstructed map freq. (%)
⬛	Background	79.51	75.89
🟦	Generic Escarpment	2.66	4.89
🟧	Continental Shelf Scarp	0.32	0.16
🟩	Canyon Scarp	3.90	3.62
🟫	Channel Escarpment	1.29	1.42
🟥	Terrace Escarpment	0.55	0.21
🟪	Fault scarp	0.21	0.03
🟥	Landslide scarp	5.09	7.29
🟪	Fold	0.28	0.04
⬜	Channel	3.34	4.19
🟨	Sediment waves	2.67	2.24
🟦	Bioconstruction	0.17	0.01

meaningful or physical morphological elements, are also present in the training predictions, indicating some degree of overfitting or noise. Interestingly, the class *bioconstruction* remains challenging to predict even in the training set, suggesting that this feature does not exhibit a strong or direct relationship with the depth, slope, and curvature used as input features. Conversely, the easiest features to predict are the linear structures associated with abrupt changes in seabed depth, which are more clearly expressed in the available input data.

To check if the predicted label distribution aligns with the real seabed features, Table 5.1 reports the class frequencies in the test regions for both the ground truth and the predicted map. The predicted frequencies are generally consistent with the original expert annotations. This confirms that the model does not collapse to predicting only the dominant background class.

Some more detailed test examples are shown in Figure 5.2, which displays three images for each window: the image on the left shows the morphological elements delineated by domain experts from the MaGIC project [3] (i.e., the ground truth); the center image shows the model's predictions; the image on the right shows the model's predictions after the cleaning algorithm described in section 4.2.3. It can be observed that, although the prediction is not entirely accurate, the model preserves the overall shape and structure of the elements. While some fine details may be lost or slightly shifted, the main features and general morphology remain identifiable, indicating that the model captures the essential patterns present in the data. White pixels in the images represent point in which no data was recorded.

Focusing on the center and right images of Figure 5.2 highlights the differences between the raw predictions and the cleaned output. It can be observed that the raw predictions sometimes include small, isolated blobs that do not correspond to any meaningful or physically plausible morphological feature. These are likely artifacts resulting from local misclassifications or noise in the model output. In contrast, the cleaned prediction, obtained by applying a simple post-processing algorithm that removes connected regions smaller than a certain pixel threshold, presents a more coherent and realistic segmentation. By filtering out these spurious elements, the cleaned output better aligns with the expected morphology and improves the interpretability and usability of the results.

To demonstrate the impact of using overlap between training windows, Figure 5.3 shows the test results obtained with the model trained without overlapping windows. A direct comparison highlights that the model trained with overlapping windows provides superior results: more features are correctly identified, and the predicted lineaments appear more continuous and less fragmented, meaning that using overlapping windows in training helps generalization. For this reason, only the results obtained with overlapping training windows are reported in the following sections.

In Figure 5.4 the confusion matrix is shown. In this study, we computed the confusion matrix after removing the background class, resulting in 11 labels. In the matrix, the rows correspond to the ground truth labels, while the columns represent the predicted labels, allowing for a detailed analysis of class-specific accuracies and misclassifications. The highest confusion values occur between Fault scarp and Generic Escarpment, with 57.47% of Fault scarp pixels being incorrectly labeled as Generic Escarpment. This outcome is expected because accurately identifying fault scarps typically requires knowledge of the underlying fault system or seismic data, which are not included in the input features. In the MaGIC seabed data, all geomorphic features with a detectable seafloor expression were mapped. Elongated seafloor unevenness that could be associated with a tectonic origin based on other datasets was interpreted as faults; however, in the absence of supporting data, these features were classified as Generic Escarpments. In other words, the Generic Escarpment label was also used for morphological features for which no additional information was available to confirm a fault origin.

Similarly, there are substantial misclassifications between Terrace Escarpment and Canyon Scarp (47.66%), and between Continental Shelf Scarp and Landslide scarp (58.99%). Classes such as Channel Escarpment and Terrace Escarpment are frequently misclassified as Landslide Scarp, indicating that they share similar visual or morphological features. This may suggest the difficulty the model has in distinguishing between certain geomorphological structures and points to specific class pairs that could benefit from improved labeling or model refinement.

On the other hand, the confusion matrix also shows good performance on several

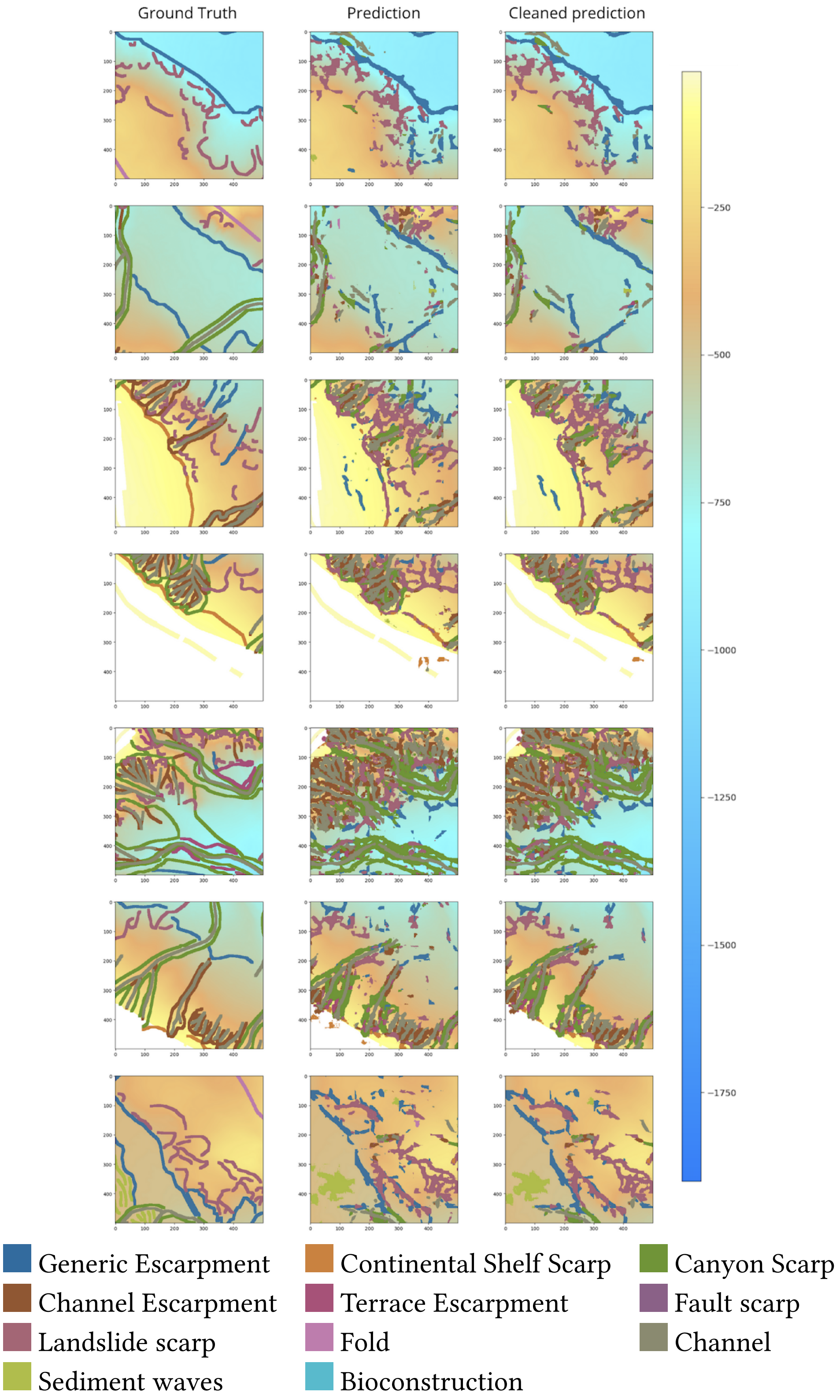


Figure 5.2: Examples of test windows. For each row: left, expert interpretation; center, model prediction; right, cleaned prediction.

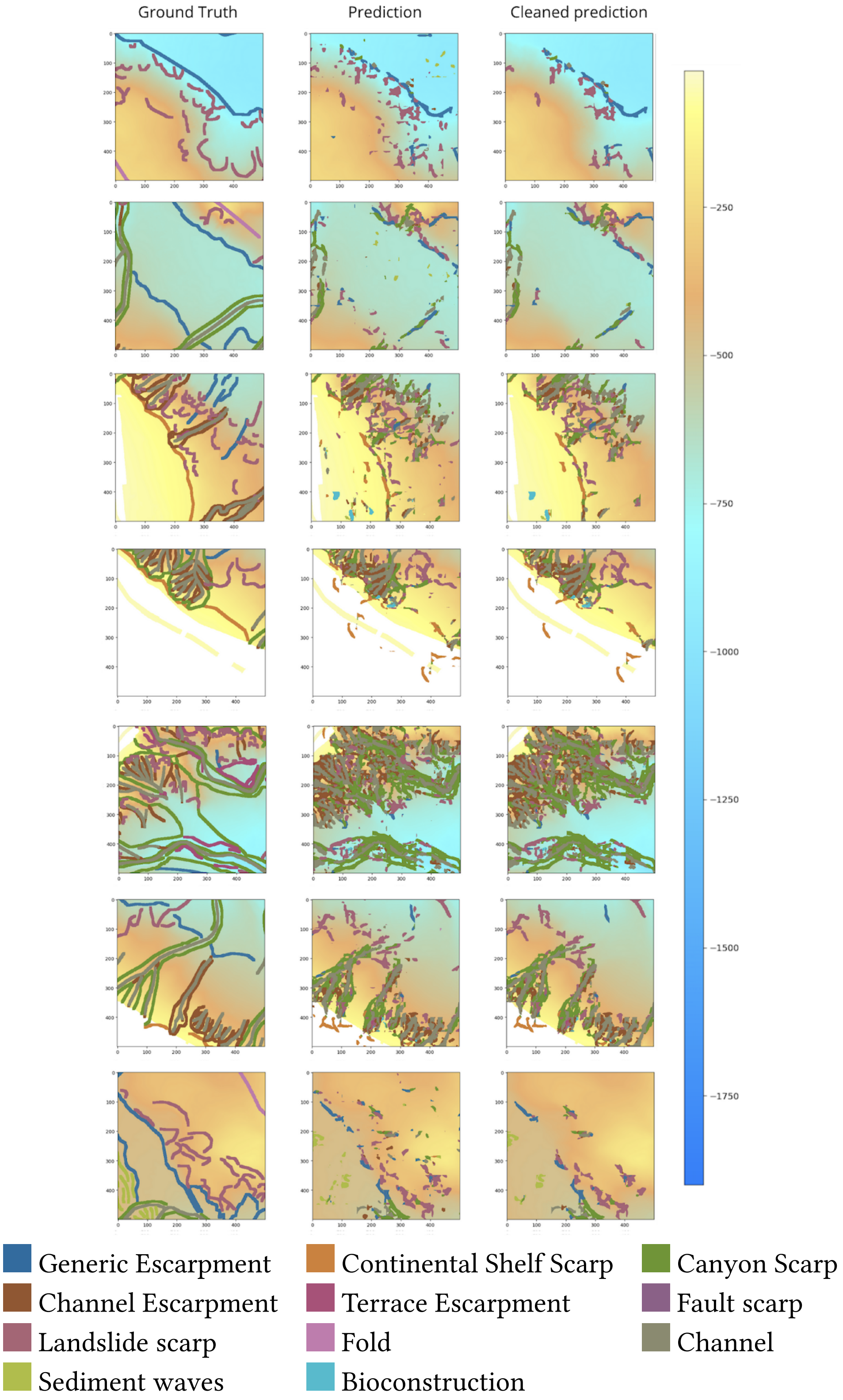
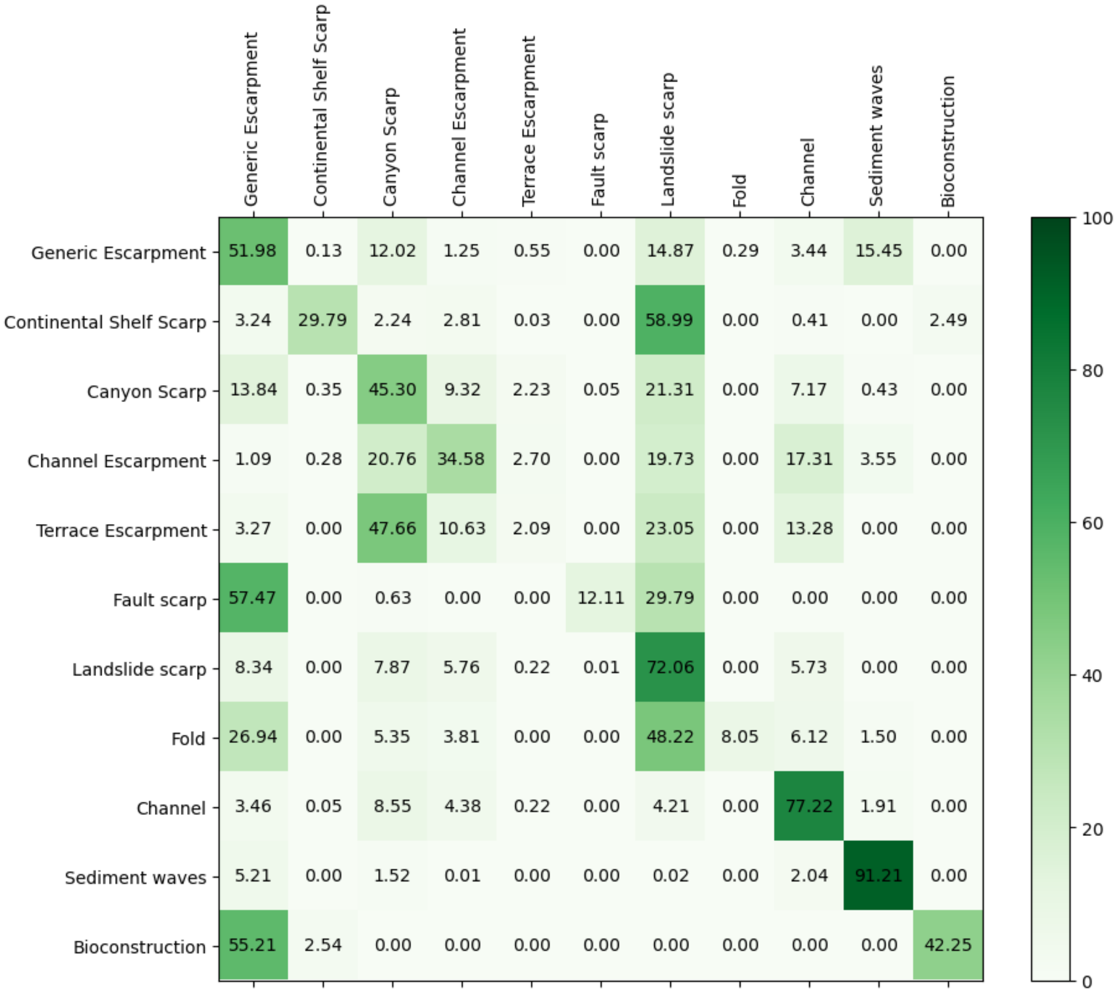


Figure 5.3: Examples of test windows without using overlapping windows during training phase. For each row: left, expert interpretation; center, model prediction; right, cleaned prediction.



	Generic Escarpment	Continental Shelf Scarp	Canyon Scarp	Channel Escarpment	Terrace Escarpment	Fault scarp	Landslide scarp	Fold	Channel	Sediment waves	Bioconstruction
Generic Escarpment	51.98	0.13	12.02	1.25	0.55	0.00	14.87	0.29	3.44	15.45	0.00
Continental Shelf Scarp	3.24	29.79	2.24	2.81	0.03	0.00	58.99	0.00	0.41	0.00	2.49
Canyon Scarp	13.84	0.35	45.30	9.32	2.23	0.05	21.31	0.00	7.17	0.43	0.00
Channel Escarpment	1.09	0.28	20.76	34.58	2.70	0.00	19.73	0.00	17.31	3.55	0.00
Terrace Escarpment	3.27	0.00	47.66	10.63	2.09	0.00	23.05	0.00	13.28	0.00	0.00
Fault scarp	57.47	0.00	0.63	0.00	0.00	12.11	29.79	0.00	0.00	0.00	0.00
Landslide scarp	8.34	0.00	7.87	5.76	0.22	0.01	72.06	0.00	5.73	0.00	0.00
Fold	26.94	0.00	5.35	3.81	0.00	0.00	48.22	8.05	6.12	1.50	0.00
Channel	3.46	0.05	8.55	4.38	0.22	0.00	4.21	0.00	77.22	1.91	0.00
Sediment waves	5.21	0.00	1.52	0.01	0.00	0.00	0.02	0.00	2.04	91.21	0.00
Bioconstruction	55.21	2.54	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	42.25

Figure 5.4: Confusion matrix computed at pixel level between predicted and ground truth maps, after removing the background class. Rows correspond to the true morphological classes, while columns indicate the predicted classes. The intensity of each cell reflects the number of pixels assigned to each pair of true and predicted classes, highlighting the agreement and confusion among the different morphological elements.

classes. In particular, Sediment Waves are correctly classified with high accuracy (91.21%), and the Channel class reaches 77.22% accuracy. Landslide Scarp, despite its visual similarity to Continental Shelf Scarp, is correctly predicted 72.06% of the time. Generic Escarpment, a potentially ambiguous class, still shows a correct classification rate of 51.98%.

Figure 5.5 shows the pixel-by-pixel success rate of each label as a function of the three input features: depth, slope, and curvature. To perform this analysis, we discretized each feature into 10 bins and computed the percentage of correctly predicted morphological elements within each bin.The results indicate that the success rate increases as the value

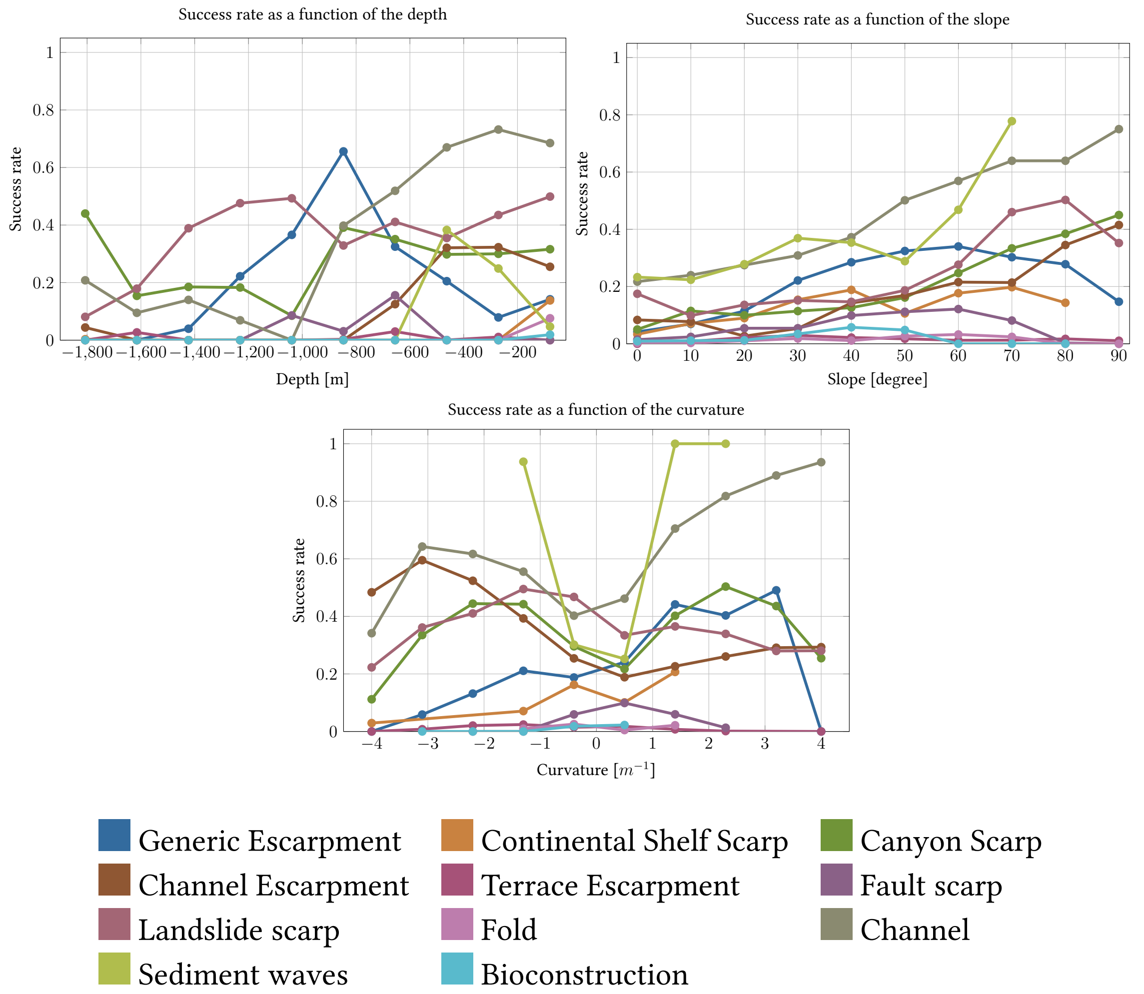


Figure 5.5: The three panels show the pixel-by-pixel success rate of each label as a function of the three input features: bathymetric depth (left), slope (right), and curvature (below). The curves correspond to the 11 morphological classes defined in the legend below.

Table 5.2: Values of the complementary metric for each morphological element computed on the predicted test windows , i.e. the center column of Figure 5.2. Two coefficients per element are computed, one for each direction of comparison.

Color	Label	$c_{\text{gt-rec}}$	$c_{\text{rec-gt}}$
🟦	Generic Escarpment	0.76	0.36
🟧	Continental Shelf Scarp	0.84	0.63
🟩	Canyon Scarp	0.81	0.71
🟫	Channel Escarpment	0.91	0.80
🟪	Terrace Escarpment	0.41	0.21
🟪	Fault scarp	0.46	0.80
🟥	Landslide scarp	0.96	0.76
🟪	Fold	0.22	0.46
⬛	Channel	0.88	0.76
🟨	Sediment waves	0.92	0.76
🟦	Bioconstruction	0.27	0.89
–	*Mean (total)*	*0.85*	*0.67*

of the slope angle becomes higher, suggesting that it is easier for the model to learn and recognize elements where the depth of the seabed change very quickly. This observation is also supported by the third plot, where the success rate generally increases when the absolute value of the curvature deviates from zero, indicating that the model performs better in areas where the seabed is not flat. Finally, we also observe that in general the success rates tend to be lower in deeper areas.

5.1.1 Metrics

We first consider the Dice coefficient, defined as $1 - L_{\text{dice}}$, which ranges from 0 to 1, with higher values indicating better overlap between prediction and ground truth. For the test windows, the Dice coefficient is approximately $c_{\text{dice}} \approx 0.8$.

However, as written in Section 4.2.4 standard metrics such as the Dice coefficient and Cross-Entropy loss do not reflect the performances of the model.

Then we compute the coefficients c described in Algorithm 1. For the test windows (500×500 pixels) we set $\epsilon = 30$ pixels to account for small spatial misalignments between the predicted and ground truth features, while still requiring a reasonably close match. The resulting scores are:

$$c_{\text{gt-rec}} = 0.79, \quad c_{\text{rec-gt}} = 0.69. \tag{5.1}$$

In Table 5.2 and Table 5.3, the coefficients for each label are reported, computed from

Table 5.3: Values of the complementary metric for each morphological element computed on the cleaned post-processed test windows of Figure 5.2. Two coefficients per element are computed, one for each direction of comparison.

Color	**Label**	$c_{\text{gt-rec}}$	$c_{\text{rec-gt}}$
🟦	Generic Escarpment	0.68	0.37
🟧	Continental Shelf Scarp	0.74	0.75
🟩	Canyon Scarp	0.78	0.72
🟫	Channel Escarpment	0.91	0.81
🟪	Terrace Escarpment	0.38	0.22
🟪	Fault Scarp	0.46	0.77
🟥	Landslide Scarp	0.91	0.78
🟪	Fold	0.23	0.83
■	Channel	0.85	0.79
🟨	Sediment Waves	0.75	0.77
🟦	Bioconstruction	0.27	1.00
–	*Mean (total)*	*0.79*	*0.69*

the predicted test windows and from the corresponding post-processed maps, respectively. As expected, the post-processing step leads to a decrease in $c_{\text{gt-rec}}$ and an increase in $c_{\text{rec-gt}}$. This behavior is reasonable, since removing isolated blobs generally reduces over-prediction, which in turn increases $c_{\text{rec-gt}}$. However, in some cases, real elements or features close to real ones may also be removed, causing a decrease in $c_{\text{gt-rec}}$. This effect is particularly evident for the Continental Shelf Scarp, Fold, and Bioconstruction classes, where the increase in $c_{\text{rec-gt}}$ is especially pronounced.

The results highlight that some elements, such as Landslide scarp and Channel, and Channel Escarpment have high scores in both directions, suggesting they are well detected by the model. In contrast, elements like Terrace Escarpment exhibit low scores, indicating that these are more challenging for the model to predict accurately.

Other elements, such as Generic Escarpment, Bioconstruction, and Fold, show an imbalance between the two coefficients. For instance, Generic Escarpments show am high $c_{\text{gt-rec}}$, indicating that where they are actually present in the ground truth, the model is able to correctly predict them with good accuracy. However, the lower $c_{\text{rec-gt}}$ suggests that the model also tends to predict Generic Escarpments in areas where they are not present, resulting in false positives. Conversely, Bioconstruction and Fold behaves in the opposite way: the high $c_{\text{rec-gt}}$ indicates that when the model predicts Bioconstruction, it is indeed present in the ground truth, meaning that these predictions are reliable. However, $c_{\text{gt-rec}}$ has low value meaning that the model often fails to predict Bioconstruction where it actually occurs, leading to false negatives. As already highlighted in Section 5.1, Bioconstruction appears to be particularly challenging to predict, even within the training areas. This suggests that the current input features are insufficient to effectively capture the patterns associated with this element, and additional or alternative features may be necessary to improve its prediction.

5.2 Simplified model configurations

Although the main focus of this thesis is the development and evaluation of a multi-class U-Net architecture for seabed morphology classification, additional experiments were carried out to investigate the behavior of simplified models. These experiments had two complementary objectives. First, they aimed to assess how the framework performs when the number of classes is reduced, thereby alleviating the strong imbalance present in the original dataset. Second, they intended to explore the limits of generalization by testing whether a model trained on extremely simplified tasks can still capture the essential characteristics of seabed morphologies.

The motivation for considering other models arises directly from the properties of the dataset. As discussed in Chapter 3, the annotated labels are dominated by the background class, with certain morphological categories representing less than 1% of the total pixels. This imbalance poses a serious challenge for any DL model, which may be biased toward predicting the majority class while ignoring underrepresented categories. Moreover, several features, such as folds or terrace escarpments, occur rarely and are difficult to delineate even for expert geologists. Under these conditions, evaluating alternative formulations of the problem becomes an important step toward understanding both the strengths and limitations of the proposed approach.

Two variants were tested in addition to the *main multi-class model*:

- a simplified classification model, where several classes were grouped to reduce sparsity and imbalance
- a binary classification model, where the task was reduced to the detection of lineaments versus background.

The results of these experiments provide valuable insight into the role of class granularity and imbalance in seabed mapping, and are discussed in the following subsections.

5.2.1 Reduced-class model

The first alternative formulation consisted of a simplified classification task in which the original set of 11 morphological categories was reduced to a smaller group. The decision to merge classes was guided by two considerations: geological similarity and statistical representation. Morphological elements that were rare and difficult to distinguish in

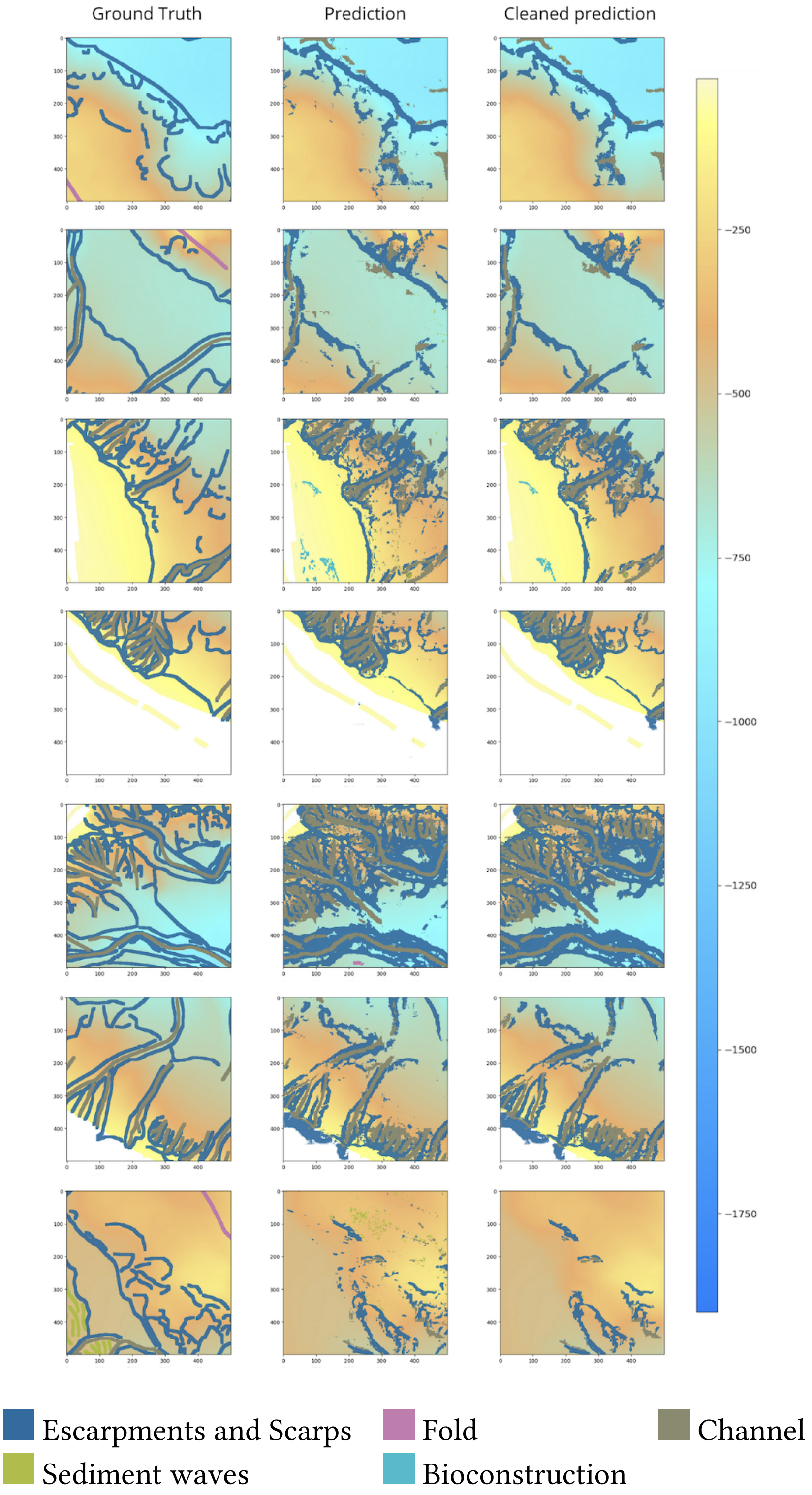


Figure 5.6: Examples of test windows. For each row: left, expert interpretation; center, model prediction; right, cleaned prediction.

practice were aggregated with related categories, producing a set of classes that were both more balanced and easier to interpret. This specific grouping is shown in Table 4.2. In this version, the first seven classes used in the *main multi-class model* (listed in Table 3.1), which include the different types of escarpments and scarps, are merged into a single class. All the other classes remain the same.

From a methodological perspective, the same U-Net architecture described in Section 4.1 was employed, with identical preprocessing, training, and post-processing procedures. The only modification was the number of output channels, corresponding to the reduced number of classes.

Figure 5.6 shows results of the experiment after training only on test areas. The first column displays the ground truth, while the second and the third shows the prediction of the model and the post-processed images respectively. The post-process shown in the figure corresponds to the cleaning procedure described in Section 4.2.3 with $n = 400$ pixels. The overall structure of the morphological lineaments is preserved. Generic escarpment is the most frequent and also the most detected element, while sediment waves are very difficult to capture.

Table 5.4: Complementary metric values for each morphological element computed for the model output and for two levels of post-processing only on test areas. The first pair of columns (*No cleaning*) refers to the raw model predictions, while the following pairs (*Cleaning* 200 *px* and *Cleaning* 400 *px*) correspond to the post-processed maps where isolated blobs smaller than 200 and 400 pixels were removed, respectively.

Color	**Label**	**No cleaning**		**Cleaning (200 px)**		**Cleaning (400 px)**	
		$c_{\text{gt-rec}}$	$c_{\text{rec-gt}}$	$c_{\text{gt-rec}}$	$c_{\text{rec-gt}}$	$c_{\text{gt-rec}}$	$c_{\text{rec-gt}}$
🟦	Escarpments and Scarps	0.90	0.94	0.85	0.95	0.82	0.96
🟪	Fold	0.08	0.08	0.08	0.09	0.08	0.18
🟫	Channel	0.91	0.69	0.88	0.72	0.87	0.73
🟩	Sediment Waves	0.84	0.60	0.25	0.61	0.08	0.60
🟦	Bioconstruction	0.78	0.03	0.00	0.00	0.00	0.00
–	*Mean (total)*	*0.88*	*0.87*	*0.76*	*0.89*	*0.72*	*0.90*

The evaluation of this model revealed several remarkable trends. The performance of the *reduced-class model* computed on post-processed output are measured using the complementary metric, that for test windows are:

$$c_{\text{gt-rec}} = 0.72, \quad c_{\text{rec-gt}} = 0.90. \tag{5.2}$$

as reported in Table 5.4.

This revels a great improvement in particular for $c_{\text{rec-gt}}$ compared to Equation (5.1) that could be interpreted as an improvement in the *precision* of the model, because, as written in Section 4.2.4, it quantify the proportion of predicted pixels that correspond to real features in the ground truth.

The complementary metrics for each classes is reported in Table 5.4. Classes that had a reasonable representation in the dataset, such as Escarpments and Scarps and Channel, maintained strong predictive performance. Escarpments and Scarps achieves the best overall performance, with balanced values from 0.90 and 0.94 for non-cleaned windows to 0.82 and 0.96 for post-processed ones. The Channel class also maintains a satisfactory performance ($c_{gt \to rec} = 0.87$, $c_{rec \to gt} = 0.73$), although with some over-prediction, as the difference between the two coefficients suggests. This confirmed that the model was able to consistently identify these features in the test regions, although some errors of over-prediction were still observed. By contrast, classes that remained sparse even after aggregation, such as Fold, exhibited poor performance, with the model failing to recognize them altogether. A peculiar case is represented by the Bioconstructions class, which shows an evident asymmetry between the two metrics. As reported in Table 5.4, it presents a fairly good recall but very low precision ($c_{gt \to rec} = 0.78$, $c_{rec \to gt} = 0.03$) for the non-cleaned maps, indicating a strong over-prediction effect. When the cleaning procedure is applied, both values drop to zero, meaning that the model tends to predict many small clusters of Bioconstructions that do not correspond to real features and are therefore removed during post-processing. Bioconstructions remain one of the most challenging classes to predict, even when the number of classes is reduced. This further supports the idea that bathymetry is not sufficient to capture the distinctive characteristics of Bioconstructions.

As expected, Figure 5.7 shows a marked reduction in cross-class confusion, becouse of the additional grouping of the classes. In the confusion matrix of the *main multi-class model* (Figure 5.4), several morphological categories, in particular Terrace escarpment and Channel escarpment, were often misclassified as Landslide scarps or Canyon scarps, due to their similar morphological structure. In the simplified setup, this misclassification disappears by construction, leading to a more stable and interpretable output. Figure 5.7 remark the difficult of the model to predict Fold and Bioconstruction, that as already said in Section 5.1, suggests that this feature does not exhibit a strong or direct relationship with the depth, slope, and curvature used as input features.

The visual outcomes shown in Figure 5.8 and Figure 5.6 confirm the quantitative trends. In Figure 5.8, the pixel-wise success rate as a function of depth, slope, and curvature shows that accuracy peaks in areas of steep slope and high curvature—conditions typical of escarpments and channels—while performance declines in flat and deep regions. This pattern is consistent with that of the *main multi-class model* (Figure 5.5), but with smoother curves, indicating that the *reduced-class model* is less sensitive to fine-scale variations in morphology.

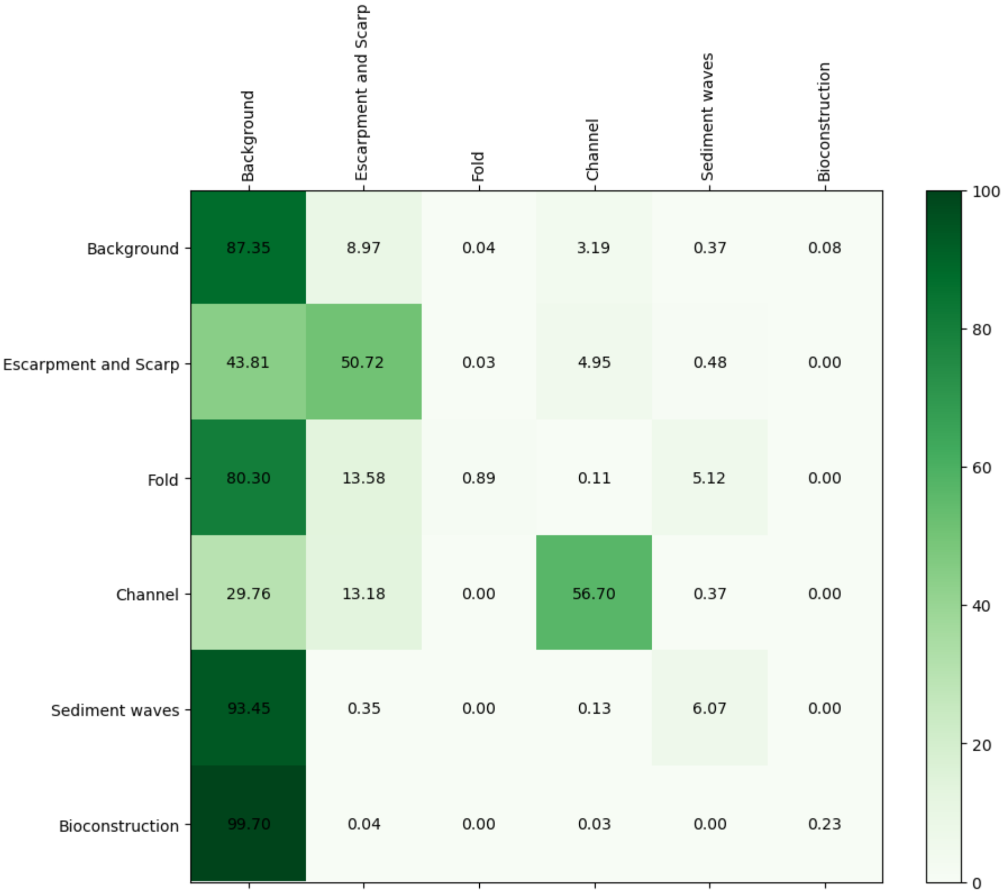


Figure 5.7: Confusion matrix computed at pixel level between predicted and ground truth maps. Rows correspond to the true morphological classes, while columns indicate the predicted classes. The intensity of each cell reflects the number of pixels assigned to each pair of true and predicted classes, highlighting the agreement and confusion among the different morphological elements

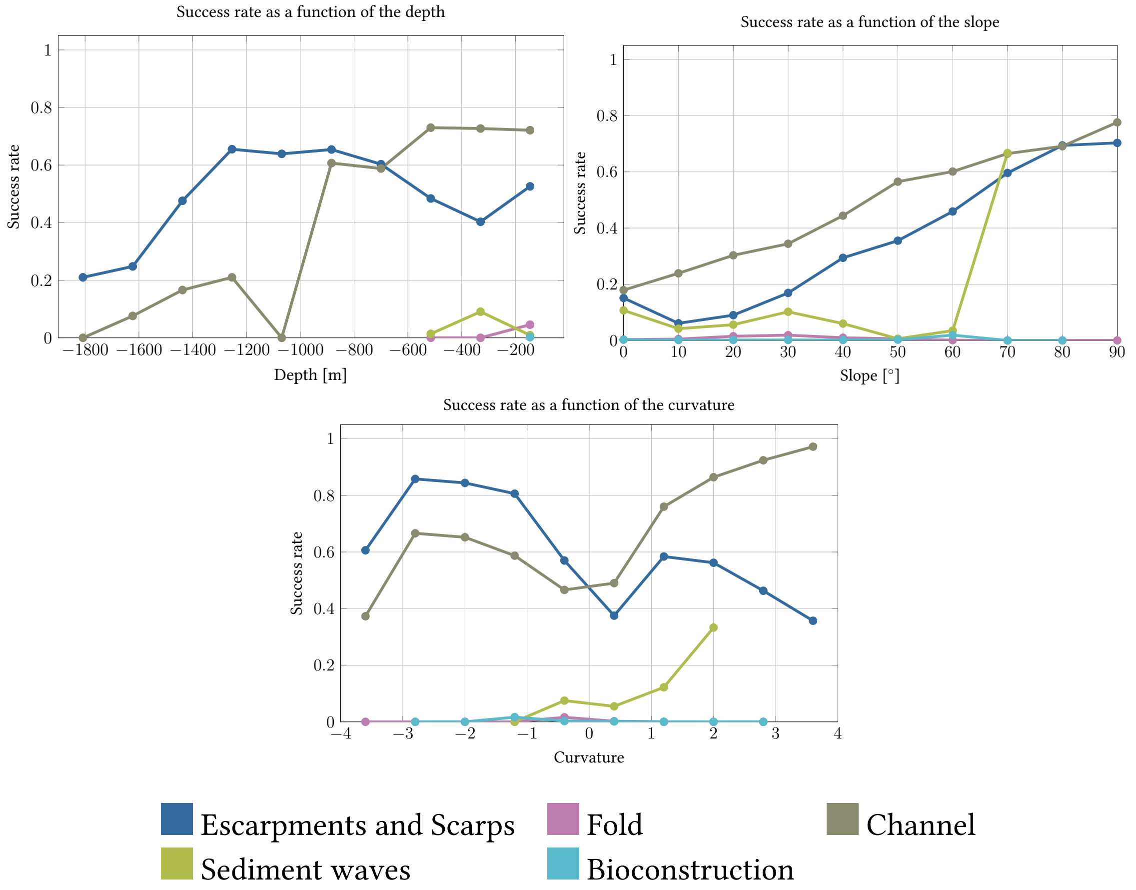


Figure 5.8: The three panels show the pixel-by-pixel success rate of each label as a function of the three input features: bathymetric depth (left), slope (right), and curvature (below). The curves correspond to the 5 morphological classes defined in the legend below.

In Figure 5.6, which compares expert interpretations, model predictions, and post-processed maps, the *reduced-class model* successfully reproduces the main morphological structure of the seabed. Escarpments and channels are delineated with good continuity, and cleaning operations effectively remove spurious patches. However, the loss of categorical diversity is visible: features such as Sediment Waves or Bioconstructions are absorbed into larger classes, reducing geological detail.

Table 5.5: Comparison between the complementary metric coefficients computed from the *main multi-class model* ($c_{\text{gt-rec-MM}}$) and from the *reduced-class* one ($c_{\text{gt-rec-RM}}$). The *reduced-class model* groups the first seven classes (various Escarpments and Scarps) into a single category.

Color	**Label**	$c_{\textbf{gt-rec-MM}}$	$c_{\textbf{gt-rec-RM}}$
🟦	Generic Escarpment	0.76	0.64
🟧	Continental Shelf Scarp	0.84	0.88
🟩	Canyon Scarp	0.81	0.95
🟫	Channel Escarpment	0.91	0.97
🟥	Terrace Escarpment	0.41	0.99
🟪	Fault Scarp	0.46	0.99
🟥	Landslide Scarp	0.96	0.97
🟪	Fold	0.22	0.08
⬛	Channel	0.88	0.91
🟨	Sediment Waves	0.92	0.84
🟦	Bioconstruction	0.27	0.78

The comparison of Table 5.5 aims to evaluate how well the *reduced-class model* reproduces the morphological elements defined in the main multi-class network. The first coefficient ($c_{\text{gt-rec-MM}}$) measures the agreement between the ground truth and the predictions of the *main multi-class model* itself, while the second one ($c_{\text{gt-rec-RM}}$) measures the same agreement but using the predictions of the *reduced-class model* instead.

By comparing these two coefficients, it is possible to estimate how much information is preserved after grouping the classes. If $c_{\text{gt-rec-RM}}$ remains close to $c_{\text{gt-rec-MM}}$, it means that the *reduced-class model* still captures the same morphological structures, even though it distinguishes fewer classes. On the other hand, a large drop indicates that some specific details of the original classes are lost, showing where the grouping causes a real loss of geological information. The comparison between $c_{\text{gt-rec-MM}}$ and $c_{\text{gt-rec-RM}}$ provides insight into how the *reduced-class model* behaves when evaluated against the original, more detailed ground truth. In the simplified network, the first seven classes (various types of Escarpments and Scarps) were grouped into a single class. As expected, these elements generally show higher values of $c_{\text{gt-rec-RM}}$ compared to $c_{\text{gt-rec-MM}}$, meaning that the *reduced-class model* is able to detect the same morphological structures more consistently, even if it no longer distinguishes between their specific subtypes. This suggests that the grouping process helps the network focus on broader, more stable patterns rather than on subtle differences that may be ambiguous or noisy in the data. Interestingly, some classes that were not part of the grouping, such as Channel and Bioconstruction, also exhibit a small improvement. This may indicate that reducing the overall number of classes simplifies the learning process and allows the model to better generalize to other morphological elements, indirectly improving their detection. In short, while the simplification leads to a loss of geological specificity, it enhances the robustness and overall interpretability of the predictions, especially for large-scale mapping applications.

When compared to the main multi-class network (Table 5.2), the *reduced-class model* shows improved performance for the Escarpment and various Scarp classes, while results for the other elements remain largely unchanged. In this case, the post-processing step can even slightly worsen the performance for some classes, likely because the *reduced-class model* already tends to over-predict less. However, this improvement comes at the cost of model specificity: by merging multiple types of Scarps and Escarpments, the model loses the ability to distinguish between their geological subtypes, resulting in a coarser but more stable representation.

5.2.2 Binary classification model

The second experiment reduces the problem to a binary task, distinguishing between lineament (any morphological feature of structural or erosional origin) and background. In this setup, all escarpments, channels, folds, and similar elements were grouped into a single positive class, labeled as Generic Lineament. The rationale behind this experiment is to test whether the model can still recognize the spatial organization of morphologically significant features when deprived of any categorical context.

Results on test areas are shown in Figure 5.9. As before, the first column displays the ground truth, the second one the predicted lineaments, while the third one displays the post-processed images. As for the previous two cases, the model preserves the main structure of the morphological lineaments, but there are no significant visual improvement respect to the *reduced-class model* of Section 5.2.1.

When contrasted with the *main multi-class model* of Section 5.1, the difference is even more pronounced. The main configuration exhibits strong class confusion, particularly between Channel escarpment and Landslide scarp, which are often misclassified in its confusion matrix. For the *binary model*, that merges all these into a single class, Figure 5.10 shows that it eliminates these ambiguities but still the pixel-wise accuracy is not that high (about 50.70). This highlight an other time that pixel-wise metric are not very indicative of the performance of the model, as explained in Section 4.2.4.

For this reason is better to look at the complementary metric. The numerical results are reported in Table 5.6. The model achieves very high complementary metrics, with

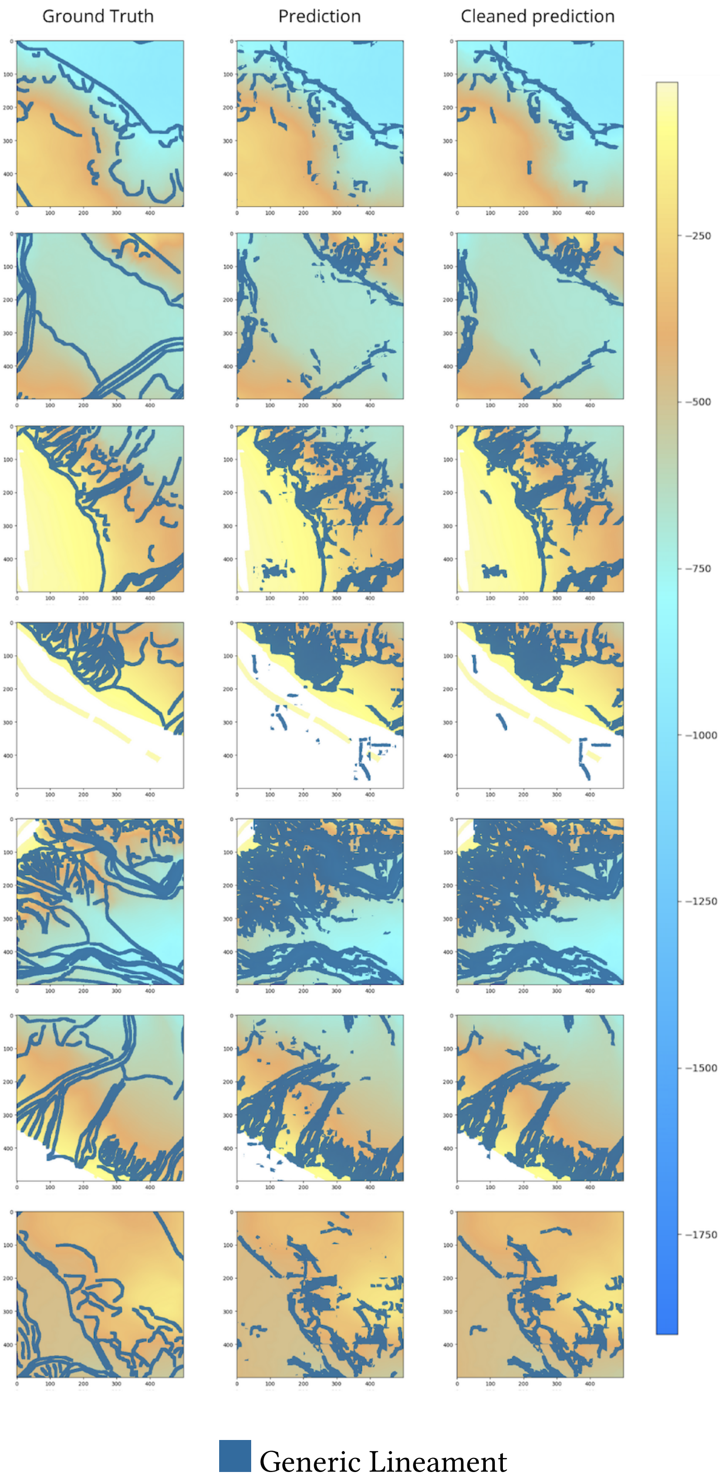


Figure 5.9: Examples of test windows for binary classification models. For each row: left, expert interpretation; center, model prediction; right, cleaned prediction.

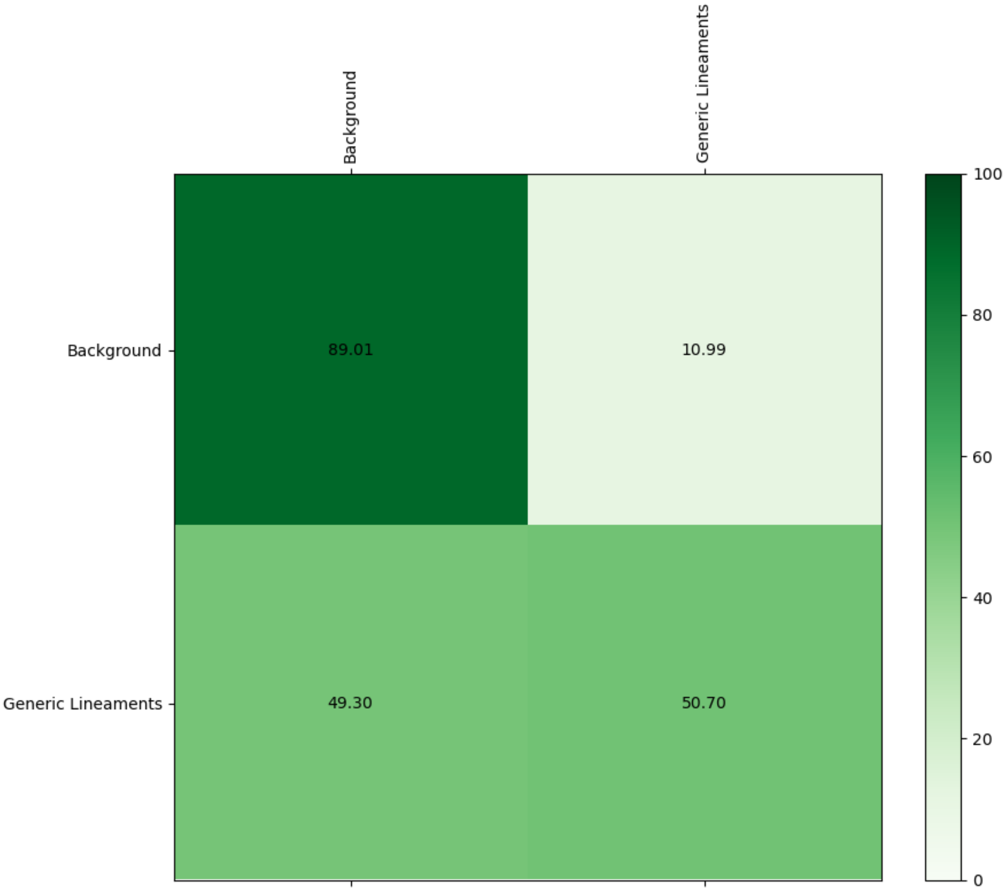


Figure 5.10: Confusion matrix computed at pixel level between predicted and ground truth maps. Rows correspond to the true morphological classes, while columns indicate the predicted classes. The intensity of each cell reflects the number of pixels assigned to each pair of true and predicted classes, highlighting the agreement and confusion among the different morphological elements

Table 5.6: Complementary metric values for the unique element computed for the model output and for two levels of post-processing only on test areas. The first pair of columns (*No cleaning*) refers to the raw model predictions, while the following pairs (*Cleaning* 200 *px* and *Cleaning* 400 *px*) correspond to the post-processed maps where isolated blobs smaller than 200 and 400 pixels were removed, respectively.

Color	**Label**	**No cleaning**		**Cleaning (200 px)**		**Cleaning (400 px)**	
		$c_{\text{gt-rec}}$	$c_{\text{rec-gt}}$	$c_{\text{gt-rec}}$	$c_{\text{rec-gt}}$	$c_{\text{gt-rec}}$	$c_{\text{rec-gt}}$
🟦	Generic Lineaments	0.90	0.95	0.85	0.96	0.81	0.97

$c_{\text{gt-rec}} = 0.90$ and $c_{\text{rec-gt}} = 0.96$ on non-cleaned windows, corresponding to the highest scores among all tested configurations. The extremely high $c_{\text{rec-gt}}$ value demonstrates that almost all predicted lineaments correspond to actual morphological features in the ground truth, while the slightly lower $c_{\text{gt-rec}}$ suggests that a small portion of real lineaments remains undetected. Compared with the *reduced-class model* of Section 5.2.1, this configuration increases both precision and recall, confirming that removing class competition allows the network to focus solely on recognizing morphological discontinuities. This is due in particular because to the absence of the classification errors between the various classes. In fact in this case we pay the price for a poor classification among the elements.

Table 5.6 also shows how the complementary metric changes after the post-processing step. As expected, $c_{\text{gt-rec}}$ decreases while $c_{\text{rec-gt}}$ tends to increase. However, the size of these variations suggests that applying post-processing in this case is unnecessary, or even counterproductive. This happens because the model is already accurate on its own and does not show significant over-prediction, so the cleaning step does not bring real benefits.

Figure 5.11 provides a more detailed view of model behavior as a function of slope and curvature. The pixel-wise success rate increases rapidly with slope, and decreases smoothly as the seabed flattens. This pattern mirrors that of both the Simplified and *main multi-class model* but with a much steeper rise, indicating that the *binary model* is more sensitive to structural discontinuities of the seabed. The same trend is observed for curvature, where positive and negative extremes—corresponding to convex ridges and concave channels—yield the best detection results.

As shown in Table 5.7, the $c_{\text{gt-rec}}$ values are generally high for almost all classes, meaning that most features present in the ground truth are correctly identified when considered as part of a single category. This confirms that the model can successfully recognize the general presence of morphological structures, even without distinguishing between their specific types. The classes that originally had weaker performance in the multi-class setup, such as Terrace Escarpment and Fault Scarp, now reach very high

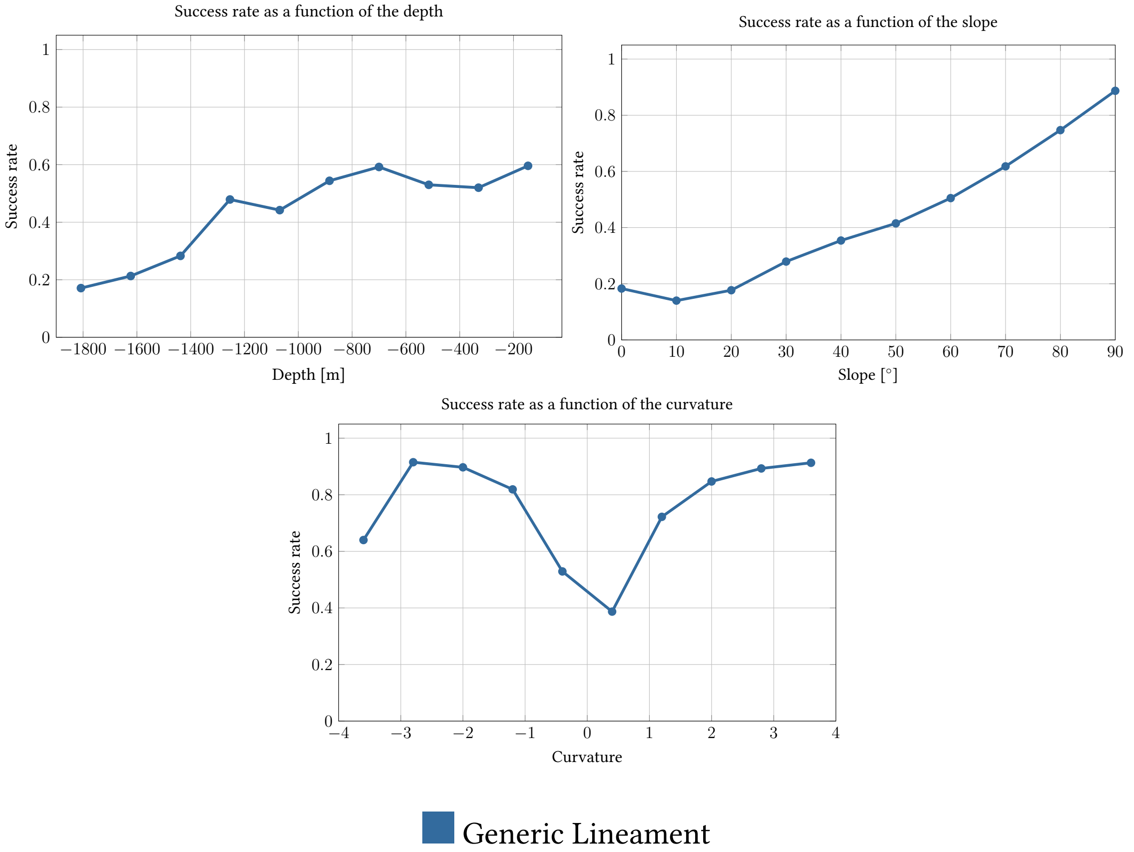


Figure 5.11: Success rate of the binary classification model as a function of bathymetric depth (top), slope (middle), and curvature (bottom). The curves confirm that the model performs best in areas characterized by strong gradients, such as steep slopes and regions of high curvature, where morphological features are more prominent. In contrast, prediction accuracy decreases in flat or deep areas, reflecting the weaker expression of lineaments in such environments.

Table 5.7: Comparison between the complementary metric coefficients computed from the *main multi-class model* ($c_{\text{gt-rec-MM}}$) and from the binary one ($c_{\text{gt-rec-BM}}$). The *binary model* groups all the classes into a single category.

Color	**Label**	$c_{\text{gt-rec-MM}}$	$c_{\text{gt-rec-BM}}$
🟦	Generic Escarpment	0.76	0.87
🟧	Continental Shelf Scarp	0.84	1.00
🟩	Canyon Scarp	0.81	0.92
🟫	Channel Escarpment	0.91	0.99
🟪	Terrace Escarpment	0.41	0.95
🟪	Fault Scarp	0.46	1.00
🟥	Landslide Scarp	0.96	0.97
🟪	Fold	0.22	0.77
⬜	Channel	0.88	0.91
🟨	Sediment Waves	0.92	0.71
🟦	Bioconstruction	0.27	1.00

scores, showing that their misclassifications were mostly due to confusion between similar structures rather than to missing detections. Even the Channel and Bioconstruction classes, which are quite different from the escarpments, maintain good performance, indicating that the binary approach captures the overall morphology of the seabed in a consistent way.

Overall, these results suggest that the *binary model* provides a robust and stable representation of the main morphological patterns, although at the cost of losing geological detail and interpretative specificity.

The binary formulation focuses exclusively on the presence or absence of morphological lineaments. This results in a model that produces maps that could serve as a first step tool for geological interpretation. From a methodological perspective, the *binary model* demonstrates that the geometric information encoded in the bathymetry, slope, and curvature channels is sufficient to describe the essential morphological structure of the seabed, even without class semantics. However, for geological applications requiring a more precise elements classification, the binary approach is still too rude.

5.3 Discussion

In this section we have displayed the main results obtained by following the training pipeline described in Chapter 4. Experiments cover three models configurations: multi-class, reduced-class, and binary. In all these cases models preserved the main geomorphic structures identified by experts, while showing expected differences in precision, generalization, and interpretative value.

The *multi-class model* (Section 5.1) represents the most detailed configuration, attempting to distinguish all eleven morphological elements. As shown in Figure 5.1 and Table 5.1, the model successfully reproduces the general spatial organization of escarpments, channels, and landslide scarps, with reconstructed frequency distributions close to those of the ground truth. The complementary metric (Table 5.2 and 5.3) highlights that classes such as Channel Escarpment and Landslide Scarp reach the highest values (up to 0.91 and 0.96 in $c_{\text{gt-rec}}$), confirming their strong geometric expression in the input maps. Conversely, Fold and Bioconstruction perform poorly ($\simeq 0.22$ and $\simeq 0.27$ in $c_{\text{gt-rec}}$), suggesting that their morphology is not well captured by depth, slope, and curvature alone. Post-processing improves overall interpretability, especially increasing values of c_{gt}. This is done by removing small spurious patches. These results are better shown in Figure 5.2 where only test windows are displayed. Anyway model still does not fully recover continuity, as linear features often remain fragmented. Despite these limitations, the multi-class model provides valuable insight into which features are most learnable and which remain ambiguous under the available data.

The *reduced-class model* (Section 5.2.1) was designed to address the strong imbalance of the dataset and the confusion observed between morphologically similar classes. Grouping the first seven escarpment-related categories into a single class produced a substantial stabilization of results. The complementary metrics reach $c_{\text{gt-rec}} = 0.72$ and $c_{\text{rec-gt}} = 0.90$ after post-processing (Equation (5.2)), indicating a significant gain in geometric precision ($c_{\text{rec-gt}}$) compared with the full model. At class level (Table 5.4), Escarpments and Scarps exhibit balanced high scores (0.82–0.96), while Channel maintains satisfactory detection (0.87/0.73). The simplification effectively eliminates cross-class ambiguities visible in the multi-class confusion matrix (Figure 5.7). As confirmed by Table 5.5, the coefficients of the reduced-class model ($c_{\text{gt-rec-RM}}$) remain close to those of the main network ($c_{\text{gt-rec-MM}}$), and even improve for several categories—for example, Terrace Escarpment and Fault Scarp rise from $0.41/0.46$ to 0.99. Figure 5.8 further shows that accuracy peaks at steep slopes and high curvatures, typical of escarpments and channels, confirming that the model captures the morphological context most effectively under strong topographic gradients.

The *binary model* (Section 5.2.2) simplifies the task to detecting the presence of any lineament against the background. This configuration removes class ambiguity entirely and produces the most stable overall pattern reconstruction. Table 5.7 shows that when all features are merged into a single class, elements that previously suffered from confusion, such as Terrace Escarpment and Fault Scarp, reach very high complementary scores. The resulting maps (Figure 5.9) clearly delineate the main structural and erosional lineaments, confirming that the geometric information in the three bathymetric derivatives is sufficient to characterize the large-scale morphology of the seabed. However, this comes at the expense of interpretative specificity, as geological meaning is lost once class semantics are removed. Therefore, the binary model is most useful as a preliminary detection stage, possibly followed by finer classification.

In summary, the results show that the proposed framework, combined with a tolerance-based evaluation, helps to join quantitative model outputs with geological interpretation. The comparison among the three models clearly illustrates how class granularity influences performance and suggests that using simpler or hierarchical classification schemes can make automated seabed mapping more reliable and easier to interpret. Moreover, in all three cases, the performance of the model are significantly better in areas characterized by large depth variations and high curvature values.

5.4 Limitations

Although the results are promising, some limitations must be acknowledged. A first limitation concerns the input data. The model relies only on bathymetric derivatives—depth, slope, and curvature—which are not always sufficient to identify every type of morphological element. For example, features such as *Bioconstruction* or *Fold* cannot be easily recognized without additional sources of information, such as backscatter or seismic data.

Another limitation is related to the dataset itself. The distribution of classes is very unbalanced: the background dominates, while some classes are represented by only a few examples. As shown in the bar plots of Chapter 3, this imbalance affects the network's ability to learn the rarest classes, which tend to be underrepresented in the predictions. In addition, some categories—particularly *Generic Escarpment*—are very broad and include a variety of morphologies. In these cases, errors may not always indicate a real failure of the model but rather the ambiguity of the label itself.

The quality of the ground truth also plays an important role. Since the annotations were made manually by different experts, they inevitably include a degree of subjectivity. This variability can influence both training and evaluation, especially for very thin or discontinuous features. Lastly, the patch-based approach used for training, while useful for increasing the number of samples and keeping train and test areas separated, limits the spatial context available to the model. Some features, such as long escarpments or channel systems, extend beyond the boundaries of a single patch and might require larger input windows or multi-scale analysis to be fully captured.

Part III

Conclusions and Open Research Questions

Chapter 6

Conclusions and Future Work

This final chapter summarizes the main results of this work and reflects on their significance. It discusses the results obtained from the different experiments presented in Chapter 5 and evaluates how effectively the proposed framework meets the research objectives introduced at the beginning of the thesis. The discussion also highlights the main strengths and weaknesses of the approach, emphasizing what has been achieved and what remains challenging when applying machine learning to seabed morphology mapping.

The chapter is divided into two parts. Section 6.1 reviews the main outcomes of the study, interpreting the results in the context of marine geology and assessing the contribution of the proposed framework to automated seabed mapping. Finally, the second section 6.2 highlights possible future developments aimed at improving model performance, integrating new data sources, and extending the approach toward risk assessment for geohazards and other applications.

6.1 Conclusions

This work explored the use of DL for the automatic classification of seabed morphological elements. The goal was to understand whether a neural network could reproduce, at least in part, the interpretative process usually performed by geologists, using only bathymetric data as input. To achieve this, a U-Net–based segmentation framework was developed and applied to the high-resolution maps of the MaGIC project, which include bathymetry and its main derivatives, i.e. slope and curvature. The model was trained on labeled examples drawn by experts and evaluated on separate test areas to ensure that the results were not influenced by overlap between training and test data.

The experiments show that the model can effectively reproduce many of the main morphological structures recognized by experts. As illustrated in the comparison maps (see Figure 5.2), the predicted features are generally well aligned with the ground truth and capture the main shapes and spatial patterns of escarpments, channels, and other lineaments. The model performs particularly well in regions characterized by strong depth gradients and high curvature, where the geomorphic expression of features is clearer and easier to detect. In contrast, in flatter or deeper zones, the predictions tend to be less accurate, reflecting the natural difficulty of identifying subtle morphological elements in these conditions.

The analysis of the quantitative results, including the confusion matrix and the bidirectional metric shown in Chapter 5, helps clarify the model's strengths and weaknesses. Certain classes, such as *Landslide Scarp* and *Channel*, are detected with good reliability, while others, such as *Bioconstruction* and *Fold*, remain challenging. These differences are mostly due to the nature of the input data: some features, like folds or biological constructions, do not have sharp topographic signatures and are difficult to recognize even by experts.

An important part of this work was the development of a complementary metric, described in Algorithm 1, designed to better reflect geological reasoning. Unlike traditional pixel-wise metrics such as Cross Entropy, this metric allows small positional differences between prediction and ground truth, thus providing a more realistic measure of success for thin and discontinuous lineaments. The experiments also included different class configurations. Besides the full multi-class model, reduced-class and binary versions were tested. In the reduced-class case, morphologically similar classes were merged, leading to more stable and balanced predictions. The binary version, which only distinguishes between background and lineament, confirmed that the model can successfully highlight structural features even in simplified conditions. Finally, the analysis of model behavior as a function of slope, curvature, and depth provided useful insights (see Figure 5.5). The success rate increases with slope and curvature, showing that the network is most effective in areas where the seabed presents clear and sharp variations.

Overall, the proposed method does not aim to replace expert interpretation but rather to support it. The model offers consistent, reproducible results that can reduce the time spent on manual mapping and help experts focus on validation and detailed geological interpretation. In this sense, it represents a valuable tool for accelerating seabed analysis while maintaining scientific reliability.

6.2 Open Research Directions

The results obtained in this work open several directions for future research. A first and natural step would be to include additional types of data in the input, such as acoustic backscatter or seismic reflection profiles. Combining these sources with bathymetry could help the model recognize classes that are not well expressed in morphology alone. Another important improvement would be to address the problem of class imbalance. Techniques such as targeted data augmentation, class reweighting, or the inclusion of new labeled areas could help the network learn rare morphological elements more effectively.

From a methodological perspective, the evaluation process could also be refined. The bidirectional metric introduced in this work has proved useful, but it could be extended to include information about the continuity and orientation of lineaments, aspects that are essential for geological interpretation. Integrating such geometry-aware metrics directly into the training process as a loss function could lead to better learning of morphological patterns. Moreover, the experiments on grouped and binary configurations suggest that multi-scale or hierarchical classification could be a promising direction. A model could first detect generic lineaments and then specialize in identifying specific types, reducing confusion between similar classes.

Finally, an exciting prospect is the use of this framework for geohazard assessment. Once the main morphological elements are automatically mapped, the model outputs could be used to identify areas prone to instability or mass movements, supporting both scientific research and offshore risk management.

In conclusion, this thesis demonstrates that DL can effectively assist in the interpretation of seabed morphology. While several challenges remain, the approach presented here provides a solid foundation for the integration of artificial intelligence into marine geology, paving the way for faster, more consistent, and more objective seabed mapping in future studies.

Acknowledgements

The research reported in this work was supported by OGS and CINECA under HPC-TRES program award number 2022-07

List of Figures

List of Tables

Bibliography

[1] Francesco L. Chiocci, Antonio Cattaneo, and Roger Urgeles. Seafloor mapping for geohazard assessment: state of the art. *Marine Geophysical Research*, 32(1):1–11, 2011. ISSN 1573-0581. doi:10.1007/s11001-011-9139-8. URL https://doi.org/10.1007/s11001-011-9139-8.

[2] Francesco L. Chiocci and Domenico Ridente. Regional-scale seafloor mapping and geohazard assessment: The experience from the italian project magic (marine geohazards along the italian coasts). *Marine Geophysical Research*, 32(1):13–23, 2011. ISSN 1573-0581. doi:10.1007/s11001-011-9120-6. URL https://doi.org/10.1007/s11001-011-9120-6.

[3] Progetto MaGIC - Marine Geohazards along the Italian Coasts, 2007-2012. URL https://www.protezionecivile.gov.it/en/approfondimento/progetto-magic-marine-geohazards-along-italian-coasts-0/. Accessed on 2024-04-15.

[4] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12):6999–7019, 2022. doi:10.1109/TNNLS.2021.3084827.

[5] Moez Krichen. Convolutional neural networks: A survey. *Computers*, 12(8), 2023. ISSN 2073-431X. doi:10.3390/computers12080151. URL https://www.mdpi.com/2073-431X/12/8/151.

[6] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. ISBN 978-3-319-24574-4. doi:10.1007/978-3-319-24574-4_28.

[7] Keir Colbo, Tetjana Ross, Craig Brown, and Tom Weber. A review of oceanographic applications of water column data from multibeam echosounders. *Estuarine, Coastal and Shelf Science*, 145:41–56, 2014. ISSN

0272-7714. doi:https://doi.org/10.1016/j.ecss.2014.04.002. URL https://www.sciencedirect.com/science/article/pii/S0272771414000900.

[8] P.K. Pandian, J.P. Ruscoe, M. Shields, J.C. Side, R.E. Harris, S.A. Kerr, and C.R. Bullen. Seabed habitat mapping techniques: an overview of the performance of various systems. *Mediterranean Marine Science*, 10(2):29–44, Dec. 2009. doi:10.12681/mms.107. URL https://ejournals.epublishing.ekt.gr/index.php/hcmr-med-mar-sc/article/view/12080.

[9] A.J Kenny, I Cato, M Desprez, G Fader, R.T.E Schüttenhelm, and J Side. An overview of seabed-mapping technologies in the context of marine habitat classification. *ICES Journal of Marine Science*, 60(2):411–418, 01 2003. ISSN 1054-3139. doi:10.1016/S1054-3139(03)00006-7. URL https://doi.org/10.1016/S1054-3139(03)00006-7.

[10] Gabriel Loureiro, André Dias, José Almeida, Alfredo Martins, Sup Hong, and Eduardo Silva. A survey of seafloor characterization and mapping techniques. *Remote Sensing*, 16(7), 2024. ISSN 2072-4292. doi:10.3390/rs16071163. URL https://www.mdpi.com/2072-4292/16/7/1163.

[11] Vicent Renard and Jean-Paul Allenou. Sea beam, multi-beam echo-sounding in "jean charcot" - description, evaluation and first results. *The International Hydrographic Review*, 56(1), Jul. 2015. URL https://journals.lib.unb.ca/index.php/ihr/article/view/23653.

[12] Xavier Lurton. *An introduction to underwater acoustics: principles and applications*. Springer Science & Business Media, 2002.

[13] Aaron Micallef, Timothy P. Le Bas, Veerle A.I. Huvenne, Philippe Blondel, Veit Hühnerbach, and Alan Deidun. A multi-method approach for benthic habitat mapping of shallow coastal areas with high-resolution multibeam data. *Continental Shelf Research*, 39-40:14–26, 2012. ISSN 0278-4343. doi:https://doi.org/10.1016/j.csr.2012.03.008. URL https://www.sciencedirect.com/science/article/pii/S0278434312000726.

[14] Markus Diesing, Sophie L. Green, David Stephens, R. Murray Lark, Heather A. Stewart, and Dayton Dove. Mapping seabed sediments: Comparison of manual, geostatistical, object-based image analysis and machine learning approaches. *Continental Shelf Research*, 84:107–119, 2014. ISSN 0278-4343. doi:https://doi.org/10.1016/j.csr.2014.05.004. URL https://www.sciencedirect.com/science/article/pii/S0278434314001629.

[15] Khaira Ismail, Veerle A.I. Huvenne, and Douglas G. Masson. Objective automated classification technique for marine landscape mapping in

submarine canyons. *Marine Geology*, 362:17–32, 2015. ISSN 0025-3227. doi:https://doi.org/10.1016/j.margeo.2015.01.006. URL `https://www.sciencedirect.com/science/article/pii/S0025322715000080`.

[16] Dick G. Simons and Mirjam Snellen. A bayesian approach to seafloor classification using multi-beam echo-sounder backscatter data. *Applied Acoustics*, 70(10):1258–1268, 2009. ISSN 0003-682X. doi:https://doi.org/10.1016/j.apacoust.2008.07.013. URL `https://www.sciencedirect.com/science/article/pii/S0003682X08001813`. The Application of Underwater Acoustics for Seabed Habitat Mapping.

[17] Evangelos Alevizos, Mirjam Snellen, Dick G. Simons, Kerstin Siemes, and Jens Greinert. Acoustic discrimination of relatively homogeneous fine sediments using bayesian classification on mbes data. *Marine Geology*, 370:31–42, 2015. ISSN 0025-3227. doi:https://doi.org/10.1016/j.margeo.2015.10.007. URL `https://www.sciencedirect.com/science/article/pii/S0025322715300475`.

[18] Timo C. Gaida, Tengku Afrizal Tengku Ali, Mirjam Snellen, Alireza Amiri-Simkooei, Thaiënne A. G. P. Van Dijk, and Dick G. Simons. A multispectral bayesian classification method for increased acoustic discrimination of seabed sediments using multi-frequency multibeam backscatter data. *Geosciences*, 8(12), 2018. ISSN 2076-3263. doi:10.3390/geosciences8120455. URL `https://www.mdpi.com/2076-3263/8/12/455`.

[19] T.C. Gaida. Acoustic mapping and monitoring of the seabed: From single-frequency to multispectral multibeam backscatter. 2020. doi:10.4233/uuid:52b8e925-b619-45f8-9056-39454e82fe02.

[20] Mirjam Snellen, Timo C. Gaida, Leo Koop, Evangelos Alevizos, and Dick G. Simons. Performance of multibeam echosounder backscatter-based classification for monitoring sediment distributions using multitemporal large-scale ocean data sets. *IEEE Journal of Oceanic Engineering*, 44(1):142–155, 2019. doi:10.1109/JOE.2018.2791878.

[21] Ed Lawson, Denson Smith, Donald Sofge, Paul Elmore, and Frederick Petry. Decision forests for machine learning classification of large, noisy seafloor feature sets. *Computers & Geosciences*, 99:116–124, 2017. ISSN 0098-3004. doi:https://doi.org/10.1016/j.cageo.2016.10.013. URL `https://www.sciencedirect.com/science/article/pii/S0098300416305842`.

[22] Xue Ji, Bisheng Yang, and Qiuhua Tang. Seabed sediment classification using multibeam backscatter data based on the selecting optimal random forest model. *Applied Acoustics*, 167:107387, 2020. ISSN 0003-682X. doi:https://doi.org/10.1016/j.apacoust.2020.107387. URL `https://www.sciencedirect.com/science/article/pii/S0003682X20304916`.

[23] Ting Zhao, Giacomo Montereale Gavazzi, Srđan Lazendić, Yuxin Zhao, and Aleksandra Pižurica. Acoustic seafloor classification using the weyl transform of multibeam echosounder backscatter mosaic. *Remote Sensing*, 13(9), 2021. ISSN 2072-4292. doi:10.3390/rs13091760. URL `https://www.mdpi.com/2072-4292/13/9/1760`.

[24] America Zelada Leon, Veerle A.I. Huvenne, Noëlie M.A. Benoist, Matthew Ferguson, Brian J. Bett, and Russell B. Wynn. Assessing the repeatability of automated seafloor classification algorithms, with application in marine protected area monitoring. *Remote Sensing*, 12(10), 2020. ISSN 2072-4292. doi:10.3390/rs12101572. URL `https://www.mdpi.com/2072-4292/12/10/1572`.

[25] Xue Ji, Bisheng Yang, and Qiuhua Tang. Acoustic seabed classification based on multibeam echosounder backscatter data using the pso-bp-adaboost algorithm: A case study from jiaozhou bay, china. *IEEE Journal of Oceanic Engineering*, 46(2): 509–519, 2021. doi:10.1109/JOE.2020.2989853.

[26] Oscar Beijbom, Peter J. Edmunds, David I. Kline, B. Greg Mitchell, and David Kriegman. Automated annotation of coral reef survey images. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1170–1177, 2012. doi:10.1109/CVPR.2012.6247798.

[27] Sébastien Villon, Marc Chaumont, Gérard Subsol, Sébastien Villéger, Thomas Claverie, and David Mouillot. Coral reef fish detection and recognition in underwater videos by supervised machine learning: Comparison between deep learning and hog+svm methods. In Jacques Blanc-Talon, Cosimo Distante, Wilfried Philips, Dan Popescu, and Paul Scheunders, editors, *Advanced Concepts for Intelligent Vision Systems*, pages 160–171, Cham, 2016. Springer International Publishing. ISBN 978-3-319-48680-2. doi:10.1007/978-3-319-48680-2_15.

[28] Hassan Mohamed, Kazuo Nadaoka, and Takashi Nakamura. Towards benthic habitat 3d mapping using machine learning algorithms and structures from motion photogrammetry. *Remote Sensing*, 12(1), 2020. ISSN 2072-4292. doi:10.3390/rs12010127. URL `https://www.mdpi.com/2072-4292/12/1/127`.

[29] Thomas Blaschke, Geoffrey J. Hay, Maggi Kelly, Stefan Lang, Peter Hofmann, Elisabeth Addink, Raul Queiroz Feitosa, Freek van der Meer, Harald van der Werff, Frieke van Coillie, and Dirk Tiede. Geographic object-based image analysis – towards a new paradigm. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87:180–191, 2014. ISSN 0924-2716. doi:https://doi.org/10.1016/j.isprsjprs.2013.09.014. URL `https://www.sciencedirect.com/science/article/pii/S0924271613002220`.

[30] Mohammad D. Hossain and Dongmei Chen. Segmentation for object-based image analysis (obia): A review of algorithms and challenges from remote sensing perspective. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150:115–134, 2019. ISSN 0924-2716. doi:https://doi.org/10.1016/j.isprsjprs.2019.02.009. URL `https://www.sciencedirect.com/science/article/pii/S0924271619300425`.

[31] Riccardo Arosio, Brandon Hobley, Andrew J Wheeler, Fabio Sacchetti, Luis A Conti, Thomas Furey, and Aaron Lim. Fully convolutional neural networks applied to large-scale marine morphology mapping. *Frontiers in Marine Science*, 10: 1228867, 2023. doi:10.3389/fmars.2023.1228867.

[32] Markus Diesing and Terje Thorsnes. Mapping of cold-water coral carbonate mounds based on geomorphometric features: An object-based approach. *Geosciences*, 8(2), 2018. ISSN 2076-3263. doi:10.3390/geosciences8020034. URL `https://www.mdpi.com/2076-3263/8/2/34`.

[33] Luis A. Conti, Aaron Lim, and Andrew J. Wheeler. High resolution mapping of a cold water coral mound. *Scientific Reports*, 9, 2019. ISSN 2045-2322. doi:10.1038/s41598-018-37725-x. URL `https://doi.org/10.1038/s41598-018-37725-x`.

[34] Larissa Macedo Cruz De Oliveira, Aaron Lim, Luis A Conti, and Andrew J Wheeler. 3d classification of cold-water coral reefs: a comparison of classification techniques for 3d reconstructions of cold-water coral reefs and seabed. *Frontiers in Marine Science*, 8:640713, 2021. doi:10.3389/fmars.2021.640713. URL `https://doi.org/10.3389/fmars.2021.640713`.

[35] Gerard Summers, Aaron Lim, and Andrew J. Wheeler. A scalable, supervised classification of seabed sediment waves using an object-based image analysis approach. *Remote Sensing*, 13(12), 2021. ISSN 2072-4292. doi:10.3390/rs13122317. URL `https://www.mdpi.com/2072-4292/13/12/2317`.

[36] Daniel Ierodiaconou, Alexandre CG Schimel, David Kennedy, Jacquomo Monk, Grace Gaylard, Mary Young, Markus Diesing, and Alex Rattray. Combining pixel and object based image analysis of ultra-high resolution multibeam bathymetry and backscatter for habitat mapping in shallow marine waters. *Marine Geophysical Research*, 39:271–288, 2018. doi:https://doi.org/10.1007/s11001-017-9338-z.

[37] Michelle Linklater, Timothy C. Ingleton, Michael A. Kinsela, Bradley D. Morris, Katie M. Allen, Michael D. Sutherland, and David J. Hanslow. Techniques for classifying seabed morphology and composition on a subtropical-temperate continental shelf. *Geosciences*, 9(3), 2019. ISSN 2076-3263. doi:10.3390/geosciences9030141. URL `https://www.mdpi.com/2076-3263/9/3/141`.

[38] Pedro S. Menandro, Alex C. Bastos, Geandré Boni, Lucas C. Ferreira, Fernanda V. Vieira, Ana Carolina Lavagnino, Rodrigo L. Moura, and Markus Diesing. Reef mapping using different seabed automatic classification tools. *Geosciences*, 10(2), 2020. ISSN 2076-3263. doi:10.3390/geosciences10020072. URL `https://www.mdpi.com/2076-3263/10/2/72`.

[39] Leo Koop, Mirjam Snellen, and Dick G. Simons. An object-based image analysis approach using bathymetry and bathymetric derivatives to classify the seafloor. *Geosciences*, 11(2), 2021. ISSN 2076-3263. doi:10.3390/geosciences11020045. URL `https://www.mdpi.com/2076-3263/11/2/45`.

[40] Lukasz Janowski, Karolina Trzcinska, Jaroslaw Tegowski, Aleksandra Kruss, Maria Rucinska-Zjadacz, and Pawel Pocwiardowski. Nearshore benthic habitat mapping based on multi-frequency, multibeam echosounder data using a combined object-based approach: A case study from the rowy site in the southern baltic sea. *Remote Sensing*, 10(12), 2018. ISSN 2072-4292. doi:10.3390/rs10121983. URL `https://www.mdpi.com/2072-4292/10/12/1983`.

[41] I.-Z. Gazis, T. Schoening, E. Alevizos, and J. Greinert. Quantitative mapping and predictive modeling of mn nodules' distribution from hydroacoustic and optical auv data linked by random forests machine learning. *Biogeosciences*, 15(23):7347–7377, 2018. doi:10.5194/bg-15-7347-2018. URL `https://bg.copernicus.org/articles/15/7347/2018/`.

[42] Muhammad Yasir, Arif Ur Rahman, and Moneeb Gohar. Habitat mapping using deep neural networks. *Multimedia Systems*, 27(4):679–690, 2021. doi:10.1007/s00530-020-00695-0.

[43] Xiaowen Luo, Xiaoming Qin, Ziyin Wu, Fanlin Yang, Mingwei Wang, and Jihong Shang. Sediment classification of small-size seabed acoustic images using convolutional neural networks. *IEEE Access*, 7:98331–98339, 2019. doi:10.1109/ACCESS.2019.2927366.

[44] Xiaoming Qin, Xiaowen Luo, Ziyin Wu, and Jihong Shang. Optimizing the sediment classification of small side-scan sonar images based on deep learning. *IEEE Access*, 9:29416–29428, 2021. doi:10.1109/ACCESS.2021.3052206.

[45] S. L. Phung, T. N. A. Nguyen, H. T. Le, P. B. Chapple, C. H. Ritz, A. Bouzerdoum, and L. C. Tran. Mine-like object sensing in sonar imagery with a compact deep learning architecture for scarce data. In *2019 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–7, 2019. doi:10.1109/DICTA47822.2019.8945982.

[46] Wong Liang Jie, Bharath Kalyan, Mandar Chitre, and Hari Vishnu. Polymetallic nodules abundance estimation using sidescan sonar: A quantitative approach using artificial neural network. In *OCEANS 2017 - Aberdeen*, pages 1–6, 2017. doi:10.1109/OCEANSE.2017.8084857.

[47] Liang Jie Wong, Bharath Kalyan, Mandar Chitre, and Hari Vishnu. Acoustic assessment of polymetallic nodule abundance using sidescan sonar and altimeter. *IEEE Journal of Oceanic Engineering*, 46(1):132–142, 2021. doi:10.1109/JOE.2020.2967108.

[48] Anum Aleem, Samabia Tehsin, Sumaira Kausar, and Amina Jameel. Target classification of marine debris using deep learning. *Intelligent Automation & Soft Computing*, 32(1):73–85, 2022. ISSN 2326-005X. doi:10.32604/iasc.2022.021583. URL `http://www.techscience.com/iasc/v32n1/45294`.

[49] Yongcan Yu, Jianhu Zhao, Quanhua Gong, Chao Huang, Gen Zheng, and Jinye Ma. Real-time underwater maritime object detection in side-scan sonar images based on transformer-yolov5. *Remote Sensing*, 13(18), 2021. ISSN 2072-4292. doi:10.3390/rs13183555. URL `https://www.mdpi.com/2072-4292/13/18/3555`.

[50] Haoting Zhang, Mei Tian, Gaoping Shao, Juan Cheng, and Jingjing Liu. Target detection of forward-looking sonar image based on improved yolov5. *IEEE Access*, 10:18023–18034, 2022. doi:10.1109/ACCESS.2022.3150339.

[51] Lihui Dong, Haolin Wang, Wei Song, Jianxin Xia, and Tongmu Liu. Deep sea nodule mineral image segmentation algorithm based on mask r-cnn. In *Proceedings of the ACM Turing Award Celebration Conference-China*, pages 278–284, 2021. doi:10.1145/3472634.3474302.

[52] J Timothy McClinton, Scott M White, and John M Sinton. Neuro-fuzzy classification of submarine lava flow morphology. *Photogrammetric Engineering & Remote Sensing*, 78(6):605–616, 2012. doi:https://doi.org/10.14358/PERS.78.6.605.

[53] Andrew P Valentine, Lara M Kalnins, and Jeannot Trampert. Discovery and analysis of topographic features using learning algorithms: A seamount case study. *Geophysical Research Letters*, 40(12):3048–3054, 2013. doi:https://doi.org/10.1002/grl.50615.

[54] Cyril Juliani. Automated discrimination of fault scarps along an arctic mid-ocean ridge using neural networks. *Computers & Geosciences*, 124:27–36, 2019. ISSN 0098-3004. doi:https://doi.org/10.1016/j.cageo.2018.12.010. URL `https://www.sciencedirect.com/science/article/pii/S009830041830596X`.

[55] Mark A. Lundine, Laura L. Brothers, and Arthur C. Trembanis. Deep learning for pockmark detection: Implications for quantitative seafloor characterization. *Geomorphology*, 421:108524, 2023. ISSN 0169-555X. doi:https://doi.org/10.1016/j.geomorph.2022.108524. URL `https://www.sciencedirect.com/science/article/pii/S0169555X22004172`.

[56] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. doi:10.1109/CVPR.2016.90.

[57] Eleni Kokinou and Costas Panagiotakis. Automatic pattern recognition of tectonic lineaments in seafloor morphology to contribute in the structural analysis of potentially hydrocarbon-rich areas. *Remote Sensing*, 12(10), 2020. ISSN 2072-4292. doi:10.3390/rs12101538. URL `https://www.mdpi.com/2072-4292/12/10/1538`.

[58] D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160(1):106–154, January 1962. ISSN 0022-3751. doi:10.1113/jphysiol.1962.sp006837.

[59] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[60] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, 1989. doi:10.1162/neco.1989.1.4.541.

[61] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9396–9405, 2019. doi:10.1109/CVPR.2019.00963.

[62] Dazhou Guo, Yanting Pei, Kang Zheng, Hongkai Yu, Yuhang Lu, and Song Wang. Degraded image semantic segmentation with dense-gram networks. *IEEE Transactions on Image Processing*, 29:782–795, 2020. doi:10.1109/TIP.2019.2936111.

[63] Jingru Yi, Pengxiang Wu, Menglin Jiang, Qiaoying Huang, Daniel J. Hoeppner, and Dimitris N. Metaxas. Attentive neural cell instance segmentation. *Medical Image Analysis*, 55:228–240, 2019. ISSN 1361-8415. doi:https://doi.org/10.1016/j.media.2019.05.004. URL `https://www.sciencedirect.com/science/article/pii/S1361841518308442`.

[64] Rohit Mohan and Abhinav Valada. Efficientps: Efficient panoptic segmentation. *International Journal of Computer Vision*, 129(5):1551–1579, 2021. ISSN 1573-1405. doi:10.1007/s11263-021-01445-z. URL `https://doi.org/10.1007/s11263-021-01445-z`.

[65] Nida M Zaitoun and Musbah J Aqel. Survey on image segmentation techniques. *Procedia Computer Science*, 65:797–806, 2015. doi:10.1016/j.procs.2015.09.027.

[66] Nameirakpam Dhanachandra, Khumanthem Manglem, and Yambem Jina Chanu. Image segmentation using k -means clustering algorithm and subtractive clustering algorithm. *Procedia Computer Science*, 54:764–771, 2015. ISSN 1877-0509. doi:https://doi.org/10.1016/j.procs.2015.06.090. URL `https://www.sciencedirect.com/science/article/pii/S1877050915014143`. Eleventh International Conference on Communication Networks, ICCN 2015, August 21-23, 2015, Bangalore, India Eleventh International Conference on Data

Mining and Warehousing, ICDMW 2015, August 21-23, 2015, Bangalore, India Eleventh International Conference on Image and Signal Processing, ICISP 2015, August 21-23, 2015, Bangalore, India.

[67] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3523–3542, 2022. doi:10.1109/TPAMI.2021.3059968.

[68] Jifeng Ning, Lei Zhang, David Zhang, and Chengke Wu. Interactive image segmentation by maximal similarity based region merging. *Pattern Recognition*, 43(2): 445–456, 2010. ISSN 0031-3203. doi:https://doi.org/10.1016/j.patcog.2009.03.004. URL `https://www.sciencedirect.com/science/article/pii/S0031320309000934`. Interactive Imaging and Vision.

[69] Sara Vicente, Vladimir Kolmogorov, and Carsten Rother. Graph cut based image segmentation with connectivity priors. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. doi:10.1109/CVPR.2008.4587440.

[70] Bo Peng, Lei Zhang, and Jian Yang. Iterated graph cuts for image segmentation. In *Asian Conference on Computer Vision*, pages 677–686. Springer, 2009. doi:10.1007/978-3-642-12304-7_64.

[71] R. Nock and F. Nielsen. Statistical region merging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1452–1458, 2004. doi:10.1109/TPAMI.2004.110.

[72] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979. doi:10.1109/TSMC.1979.4310076.

[73] N. Senthilkumaran and R. Rajesh. Image segmentation - a survey of soft computing approaches. In *2009 International Conference on Advances in Recent Technologies in Communication and Computing*, pages 844–846, 2009. doi:10.1109/ARTCom.2009.219.

[74] Orlando José Tobias and Rui Seara. Image segmentation by histogram thresholding using fuzzy sets. *IEEE transactions on Image Processing*, 11(12):1457–1465, 2002. doi:10.1109/TIP.2002.806231.

[75] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015. doi:10.1109/CVPR.2015.7298965.

[76] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017. doi:10.1109/TPAMI.2016.2644615.

[77] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 173–190, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58539-6. doi:10.1007/978-3-030-58539-6_11.

[78] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In Danail Stoyanov, Zeike Taylor, Gustavo Carneiro, Tanveer Syeda-Mahmood, Anne Martel, Lena Maier-Hein, João Manuel R.S. Tavares, Andrew Bradley, João Paulo Papa, Vasileios Belagiannis, Jacinto C. Nascimento, Zhi Lu, Sailesh Conjeti, Mehdi Moradi, Hayit Greenspan, and Anant Madabhushi, editors, *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 3–11, Cham, 2018. Springer International Publishing. ISBN 978-3-030-00889-5. doi:10.1007/978-3-030-00889-5_1.

[79] Özgün Çiçek, Ahmed Abdulkadir, Soeren S. Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: Learning dense volumetric segmentation from sparse annotation. In Sebastien Ourselin, Leo Joskowicz, Mert R. Sabuncu, Gozde Unal, and William Wells, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, pages 424–432, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46723-8. doi:10.1007/978-3-319-46723-8_49.

[80] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. Road extraction by deep residual u-net. *IEEE Geoscience and Remote Sensing Letters*, 15(5):749–753, 2018. doi:10.1109/LGRS.2018.2802944.

[81] Khoa Luu, Chenchen Zhu, Chandrasekhar Bhagavatula, T. Hoang Ngan Le, and Marios Savvides. A deep learning approach to joint face detection and segmentation. In Michal Kawulok, M. Emre Celebi, and Bogdan Smolka, editors, *Advances in Face Detection and Facial Image Analysis*, pages 1–12. Springer International Publishing, Cham, 2016. ISBN 978-3-319-25958-1. doi:10.1007/978-3-319-25958-1_1. URL `https://doi.org/10.1007/978-3-319-25958-1_1`.

[82] Çağrı Kaymak and Ayşegül Uçar. A brief survey and an application of semantic image segmentation for autonomous driving. In Valentina Emilia Balas, Sanjiban Sekhar Roy, Dharmendra Sharma, and Pijush Samui, editors, *Handbook of Deep Learning Applications*, pages 161–200. Springer International Publishing, Cham, 2019. ISBN 978-3-030-11479-4. doi:10.1007/978-3-030-11479-4_9. URL `https://doi.org/10.1007/978-3-030-11479-4_9`.

[83] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. 2016. doi:10.48550/arXiv.1412.7062. URL `https://arxiv.org/abs/1412.7062`.

[84] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2018. doi:10.1109/TPAMI.2017.2699184.

[85] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6230–6239, 2017. doi:10.1109/CVPR.2017.660.

[86] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3141–3149, 2019. doi:10.1109/CVPR.2019.00326.

[87] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):386–397, 2020. doi:10.1109/TPAMI.2018.2844175.

[88] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8759–8768, 2018. doi:10.1109/CVPR.2018.00913.

[89] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8810–8818, 2019. doi:10.1109/CVPR.2019.00902.

[90] Bowen Cheng, Maxwell D. Collins, Yukun Zhu, Ting Liu, Thomas S. Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12472–12482, 2020. doi:10.1109/CVPR42600.2020.01249.

[91] Stanislav Nikolov, Sam Blackwell, Ruheena Mendes, Jeffrey De Fauw, Clemens Meyer, Cían Hughes, Harry Askham, Bernardino Romera-Paredes, Alan Karthikesalingam, Carlton Chu, Dawn Carnell, Cheng Boon, Derek D'Souza, Syed Ali Moinuddin, Kevin Sullivan, DeepMind Radiographer Consortium, Hugh Montgomery, Geraint Rees, Ricky Sharma, Mustafa Suleyman, Trevor Back, Joseph R. Ledsam, and Olaf Ronneberger. Deep learning to achieve clinically applicable

segmentation of head and neck anatomy for radiotherapy. *CoRR*, abs/1809.04430, 2018. doi:10.48550/arXiv.1809.04430. URL `http://arxiv.org/abs/1809.04430`.

[92] Philipp Kickingereder, Fabian Isensee, Irada Tursunova, Jens Petersen, Ulf Neuberger, David Bonekamp, Gianluca Brugnara, Marianne Schell, Tobias Kessler, Martha Foltyn, Inga Harting, Felix Sahm, Marcel Prager, Martha Nowosielski, Antje Wick, Marco Nolden, Alexander Radbruch, Jürgen Debus, Heinz-Peter Schlemmer, Sabine Heiland, Michael Platten, Andreas von Deimling, Martin J. van den Bent, Thierry Gorlia, Wolfgang Wick, Martin Bendszus, and Klaus H. Maier-Hein. Automated quantitative tumour response assessment of mri in neuro-oncology with artificial neural networks: a multicentre, retrospective study. *The Lancet Oncology*, 20(5):728–740, 2019. doi:10.1016/S1470-2045(19)30098-1. URL `https://doi.org/10.1016/S1470-2045(19)30098-1`.

[93] Fabian Isensee, Philipp Kickingereder, Wolfgang Wick, Martin Bendszus, and Klaus H. Maier-Hein. Brain tumor segmentation and radiomics survival prediction: Contribution to the brats 2017 challenge. In Alessandro Crimi, Spyridon Bakas, Hugo Kuijf, Bjoern Menze, and Mauricio Reyes, editors, *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, pages 287–297, Cham, 2018. Springer International Publishing. ISBN 978-3-319-75238-9. doi:10.1007/978-3-319-75238-9_25.

[94] Ozan Oktay, Jo Schlemper, Loïc Le Folgoc, Matthew C. H. Lee, Mattias P. Heinrich, Kazunari Misawa, Kensaku Mori, Steven G. McDonagh, Nils Y. Hammerla, Bernhard Kainz, Ben Glocker, and Daniel Rueckert. Attention u-net: Learning where to look for the pancreas. *CoRR*, abs/1804.03999, 2018. doi:10.48550/arXiv.1804.03999. URL `http://arxiv.org/abs/1804.03999`.

[95] Nahian Siddique, Sidike Paheding, Colin P. Elkin, and Vijay Devabhaktuni. U-net and its variants for medical image segmentation: A review of theory and applications. *IEEE Access*, 9:82031–82057, 2021. ISSN 2169-3536. doi:10.1109/access.2021.3086020. URL `http://dx.doi.org/10.1109/ACCESS.2021.3086020`.

[96] Qiangguo Jin, Zhaopeng Meng, Tuan D. Pham, Qi Chen, Leyi Wei, and Ran Su. Dunet: A deformable network for retinal vessel segmentation. *Knowledge-Based Systems*, 178:149–162, 2019. ISSN 0950-7051. doi:https://doi.org/10.1016/j.knosys.2019.04.025. URL `https://www.sciencedirect.com/science/article/pii/S0950705119301984`.

[97] S. M. Kamrul Hasan and Cristian A. Linte. A modified u-net convolutional network featuring a nearest-neighbor re-sampling-based elastic-transformation for brain tissue characterization and segmentation. In *2018 IEEE Western*

New York Image and Signal Processing Workshop (WNYISPW), pages 1–5, 2018. doi:10.1109/WNYIPW.2018.8576421.

[98] Hideki Kakeya, Toshiyuki Okada, and Yukio Oshiro. 3d u-japa-net: Mixture of convolutional networks for abdominal multi-organ ct segmentation. In Alejandro F. Frangi, Julia A. Schnabel, Christos Davatzikos, Carlos Alberola-López, and Gabor Fichtinger, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, pages 426–433, Cham, 2018. Springer International Publishing. ISBN 978-3-030-00937-3. doi:10.1007/978-3-030-00937-3_49.

[99] Song LI, Geoffrey K.F. TSO, and Kaijian HE. Bottleneck feature supervised u-net for pixel-wise liver and tumor segmentation. *Expert Systems with Applications*, 145:113131, 2020. ISSN 0957-4174. doi:https://doi.org/10.1016/j.eswa.2019.113131. URL `https://www.sciencedirect.com/science/article/pii/S0957417419308486`.

[100] Han Zhang, Hongqing Zhu, and Xiaofeng Ling. Polar coordinate sampling-based segmentation of overlapping cervical cells using attention u-net and random walk. *Neurocomputing*, 383:212–223, 2020. ISSN 0925-2312. doi:https://doi.org/10.1016/j.neucom.2019.12.036. URL `https://www.sciencedirect.com/science/article/pii/S0925231219317394`.

[101] Nabil Ibtehaz and M. Sohel Rahman. Multiresunet : Rethinking the u-net architecture for multimodal biomedical image segmentation. *Neural Networks*, 121:74–87, 2020. ISSN 0893-6080. doi:https://doi.org/10.1016/j.neunet.2019.08.025. URL `https://www.sciencedirect.com/science/article/pii/S0893608019302503`.

[102] Alessia Colonna, Fabio Scarpa, and Alfredo Ruggeri. Segmentation of corneal nerves using a u-net-based convolutional neural network. In Danail Stoyanov, Zeike Taylor, Francesco Ciompi, Yanwu Xu, Anne Martel, Lena Maier-Hein, Nasir Rajpoot, Jeroen van der Laak, Mitko Veta, Stephen McKenna, David Snead, Emanuele Trucco, Mona K. Garvin, Xin Jan Chen, and Hrvoje Bogunovic, editors, *Computational Pathology and Ophthalmic Medical Image Analysis*, pages 185–192, Cham, 2018. Springer International Publishing. ISBN 978-3-030-00949-6. doi:10.1007/978-3-030-00949-6_22.

[103] D. Stevens, S. Dragicevic, and K. Rothley. icity: A gis–ca modelling tool for urban planning and decision making. *Environmental Modelling & Software*, 22(6):761–773, 2007. ISSN 1364-8152. doi:https://doi.org/10.1016/j.envsoft.2006.02.004. URL `https://www.sciencedirect.com/science/article/pii/S1364815206000582`.

[104] Lorena Parra. Remote sensing and gis in environmental monitoring. *Applied Sciences*, 12(16), 2022. ISSN 2076-3417. doi:10.3390/app12168045. URL `https://www.mdpi.com/2076-3417/12/16/8045`.

[105] Cynthia A. Brewer. Basic mapping principles for visualizing cancer data using geographic information systems (gis). *American Journal of Preventive Medicine*, 30(2, Supplement):S25–S36, 2006. ISSN 0749-3797. doi:https://doi.org/10.1016/j.amepre.2005.09.007. URL `https://www.sciencedirect.com/science/article/pii/S0749379705003582`.

[106] Silvia Ceramicola, Francesco Fanucci, Cesare Corselli, E. Colizza, Daniele Morelli, Andrea Cova, Alessandra Savini, Daniel Praeg, Massimo Zecchin, Andrea Caburlotto, Oliviero Candoni, D. Civile, Marianne Coste, Diego Cotterle, Salvatore Critelli, Angela Cuppari, Michele Deponte, R. Dominici, E. Forlin, E. Gordini, C. Tessarolo, F. Marchese, F. Muto, S. Palamara, Ramella Riccardo, Lorenzo Facchin, and Roberto Romeo. Calabria Ionica (Tavola 8, Fogli 35-39). In Chiocci FL, Budillon F, Ceramicola S, Gamberi F, and Orrù P (Eds), editors, *Atlante dei lineamenti di pericolosità geologica dei mari italiani - Risultati del progetto MaGIC*, pages 174–195. 2021. URL `https://hal.science/hal-03553256`.

[107] Silvia Ceramicola, Maria Rosaria Senatore, Andrea Cova, Agostino Meo, Massimo Zecchin, Daniel Praeg, Cotterle Diego, Critelli Salvatore, Andrea Caburlotto, Dario Civile, Marianne Coste, R Dominici, E. Forlin, F Muto, Alessandro Bosman, Chiocci Francesco Latino, E. Lai, Daniele Casalbore, E. Morelli, Oliviero Candoni, E. Gordini, Deponte Michele, Ramella Riccardo, Lorenzo Facchin, and Roberto Romeo. Golfo di Taranto (Tavola 9, Fogli 40-46). In Chiocci FL, Budillon F, Ceramicola S, Gamberi F, and Orrù P (compilers), editors, *Atlante dei lineamenti di pericolosità geologica dei mari italiani - Risultati del progetto MaGIC*, pages 196–225. CNR Edizioni, 2021. URL `https://hal.science/hal-03553376`.

[108] F.L. Chiocci, F. Budillon, S. Ceramicola, F. Gamberi, and P. Orrù, editors. *Atlante dei lineamenti di pericolosità geologica dei mari italiani-Risultati del progetto MaGIC*. CNR Edition, 2021. ISBN 978-88-8080-457-4. URL `https://hdl.handle.net/20.500.14083/6380`.

[109] Alexandre Milesi. `Pytorch-UNet`: Pytorch implementation of the u-net for image semantic segmentation with high quality images, 2024. URL `https://github.com/milesial/Pytorch-UNet`.

[110] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? 2019. doi:https://doi.org/10.48550/arXiv.1805.11604. URL `https://arxiv.org/abs/1805.11604`.

[111] Rongjian Zhao, Buyue Qian, Xianli Zhang, Yang Li, Rong Wei, Yang Liu, and Yinggang Pan. Rethinking dice loss for medical image segmentation. In *2020 IEEE International Conference on Data Mining (ICDM)*, pages 851–860, 2020. doi:10.1109/ICDM50108.2020.00094.

[112] Shruti Jadon. A survey of loss functions for semantic segmentation. In *2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–7, 2020. doi:10.1109/CIBCB48159.2020.9277638.

[113] Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with restarts. *CoRR*, abs/1608.03983, 2016. doi:https://doi.org/10.48550/arXiv.1608.03983. URL `http://arxiv.org/abs/1608.03983`.