*Article*

# Automatic Generation of a Portuguese Land Cover Map with Machine Learning

**Antonio Esteves** [1,†,*] and **Nuno Valente** [2,†,*]

[1] ALGORITMI Research Centre/LASI, University of Minho, Braga, Portugal; esteves@di.uminho.pt
[2] University of Minho, Braga, Portugal; a81986@alunos.uminho.pt
[*] Correspondence: esteves@di.uminho.pt
[†] These authors contributed equally to this work.

**Abstract:** The application of machine learning techniques to satellite imagery has been the subject of interest in recent years. The increase in quality and quantity of images, made available by Earth observation programs, such as the `Copernicus` program, led to the generation of large amounts of data. Among the various applications of this data is the creation of land cover maps. The present work aimed to create machine learning models capable of accurately segment and classify satellite images, to automatically generate a land cover map of the Portuguese territory. Several experiments were carried out with the spectral bands of the Sentinel-2 satellite, with vegetation indices, and with several sets of land cover classes. Three machine learning architectures were evaluated, which adopt two different techniques for image classification. One of the classification techniques follows an object-oriented approach, and in this case the architecture adopted in our models was a `U-Net` artificial neural network. The other classification technique is pixel-oriented, and the machine learning models tested were random forest and support vector machine. The overall accuracy of the results obtained ranged from 68.6% to 94.75%, depending strongly on the number of classes into which the land cover is classified. The result of 94.75% was obtained when classifying the land cover only into 5 classes. However, a very interesting accuracy of 92.37% was achieved by the model when trained to classify 8 classes. These results are superior to those reported in the related bibliography.

**Keywords:** machine learning; deep learning; remote sensing; land cover map

## 1. Introduction

Recent scientific advances in remote sensing (RS) have resulted in easy access to satellite imagery. Among the countless applications of satellite imagery, the present work highlights the creation of land use land cover (LULC) maps. LULC refers to human constructions and natural features of the earth's surface. LULC are used in various fields of study, such as urban planning, natural resource management, carbon circulation, epidemiology, and climate change. Using the Portuguese territory as a case study, this work intends to apply machine learning (ML) techniques to reproduce the results of the Corine Land Cover (CLC) European Union project.

One of the expected outcomes for this project was to train a model capable of successfully classifying satellite imagery into a LULC map. In a LULC classification task, the term of comparison is an overall accuracy of 85% and where none of the classes have an accuracy of less than 70% [1]. If a trained model performs better than this threshold, it will be considered successful.

RS tasks, such as LULC classification, exhibit some unique specificities. Although there are huge amounts of satellite imagery, most of this data is not classified or it is outdated, therefore not being useful for training deep learning (DL) models [2]. The seasons introduce variability, and hence complexity, especially due to changes in phenology [3]. However this variability can be captured by DL methods, provided it is reproduced in the training data [4].

Several techniques can be implemented for LULC classification, however, these can be divided into two categories: pixel-oriented and object-oriented. Pixel-oriented techniques are more traditional and consider each pixel as an independent unit, classifying each pixel according to its spectral values. Due to their technical limitations, pixel-oriented methods should not be used with high-resolution images [5], as they lower model accuracy and generate images that suffer from the salt and pepper problem, as mentioned by [6]. These methods have two additional limitations [7]:

- Cann't handle mixed pixels, a phenomenon that occurs when features from multiple classes are present in a single pixel.
- Don't take advantage of the content of adjacent pixels and their contextual information.

Object-oriented methods, also called geographic object-based image analysis (GEOBIA), group pixels into segments that ideally represent real-world objects. Typically GEOBIA takes place in two phases: segmentation and classification. Note that the segmentation process, not present in pixel-oriented techniques, can also introduce errors into the model, especially in cases of sub-segmentation [8].

## 2. Related Work

Several papers reporting the application of machine learning to remote sensing have been published recently. This section focus on related approaches to pixel- and object-based land cover classification, the employed ML models, the datasets, the considered land cover classes, and other techniques such as the inclusion of spectral indexes.

A comparison of five ML models was documented in [4]. The chosen models where Random Forest (RF) and four Convolutional Neural Networks (CNNs). These models classified Sentinel-2 imagery into 8 classes using the 4 bands with a spacial resolution of 10 meters (red, green, blue, NIR). The achieved results where compared to TOP10NL data from the Infrastructure for Spatial Information in the European Community (INSPIRE). The RF was the worse model with an overall accuracy of 81%, and the best model obtained 86% accuracy. The main conclusions from this were (i) CNNs and RFs are capable of classifying land cover classes, (ii) hyper-parameter optimization has reduced effect on results when adequate amount of training data is available, (iii) seasonal variety can be handled by introducing it into the training set, (iv) in 3 out of 4 CNN models the size of the input impacts the classification results, and (v) transfer learning shows acceptable results, making the usage of several additional data valid when the application targets an European map.

[9] presents another comparison of ML models, including Support Vector Machines (SVM), extreme gradient boosting (XGBoost), RF, and an Artificial Neural Network. Their case study was the boreal climate and the considered surface area has a dimension of $10km \times 12km$. The models were feed with four images, one per season, which improved the classification of some classes. The model with the best result was a SVM with an accuracy of 75.8%.

The work presented in [1] has objectives and methodology similar to ours and can therefore be used as a comparison term. The paper evaluates the feasibility of applying the U-Net neural network to classify the land cover. It achieved a classification accuracy of 92% using the 5 CLC level 1 classes, which decreases to 84% when 13 CLC level 2 classes are considered. The model obtains the worst results when using only RGB bands, on the contrary, the best model was obtained with a combination of spectral bands and computed spectral indexes such as NDVI.

[10] reports a successful application of a RF, over a combination of Sentinel-1 and Sentinel-2 imagery data, to crop mapping in Belgium. The Model mapped the Belgium territory in 12 classes and two steps. The first step classifies the objects in one out of 4 classes: built-up, water, forest, and crop. The second step expands the crop class into 9 more specific classes. They achieved an 82% overall accuracy.

The paper from [11], published in 2019, introduced a new large-scale dataset for training ML models to classify or segment satellite imagery. The dataset is called BigEarthNet and contains $590,326$ image patches. This significant amount of data alleviates the problem encountered in RS, the lack of large training sets, a bottleneck that prevents the use of the more recent and complex deep learning models.

## 3. Methodology

### 3.1. BigEarthNet Dataset

The main dataset we adopted to to train ML models was the mentioned BigEarthNet [11]. This dataset consists of $590,326$ Sentinel-2 image patches, composed of 12 spectral bands with 10, 20, and 60 meters of spacial resolution, and each patch is labeled with one or more CLC classes. The bands with 60 meters spacial resolution were discarded.

Although this dataset is aimed for classification problems, a layer was added to each patch, obtained from a Web Map Service, containing the corresponding 2018 CLC map. This layer was necessary to train the models for pixel-wise classification, instead of patch-wise classification. From the total of $590,326$ images, $16,110$ were removed, either because they contain clouds or because there was no CLC map available. Although the BigEarthNet dataset contains images covering the four seasons, land cover classes are not balanced. The CLC class corresponding to glaciers and perpetual snow is totally absent from the dataset.

### 3.2. LandCoverPT Dataset

The BigEarthNet dataset gathers images from several European countries, some of which have biomes drastically different from Portugal. The LandCoverPT was created with the objective of having dataset more appropriate to train ML models capable of generating a Portuguese land cover map.

The creation of the dataset used 26 Sentinel-2 products, captured in June and August 2019, and the products where divided into $153,347$ patches with the same size as the BigEarthNet patches ($120 \times 120$).

A few aspects to take in consideration when analysing the results produced with the LandCoverPT dataset:

1. It does not include seasonal variety.
2. A thorough examination to identify the presence of clouds was not carried out, and so there may be a residual amount of clouds not detected by manual inspection.
3. Some level 3 CLC classes are missing, since they do not exist in Portuguese territory.

### 3.3. Models

The first attempt to classify the land cover was carried out with a Support Vector Machine (SVM) [12] [13]. The work reported in [9] compares the SVM to the random forest, the extreme gradient boosting (XGBoost), and a deep neural network. An SVM constructs a hyperplane, in a high dimensional space, to separate each pair of classes. A good separation is achieved by the hyperplane that has the largest distance to the nearest training samples, in order to minimize the generalization error of the classifier. The separating hyperplane depends on a subset of the training data, called the support vectors. A hard margin SVM tries to fit a decision boundary that maximizes the distance between the support vectors of the two classes, but this type of SVM classifier is very sensitive to outliers and it only works on data that is linearly separable. The soft margin SVM addresses these problems by allowing some samples to be located on the boundary region. Thus, a soft margin classifier deals with a trade-off between maximizing the width of the separating margin and minimizing the misclassifications. The trade-off is controlled by the $C$ hyperparameter of scikit-learn `SVC` classifier.

In machine learning, kernels can help to construct non-linear decision boundaries using linear classifiers. A kernel function only calculates the relationships between every pair of samples as if they were in a higher dimensional space. This trick, consisting in calculating the high-dimensional relationships without actually transforming the samples to the higher dimension, is called the kernel trick. The kernel trick reduces the amount of computation required by SVMs by avoiding the transformation of the data from a lower to higher dimensional space. There are several types of kernels, such as polynomial and

Gaussian kernels. The (Gaussian) Radial Basis Function kernel, computed with the pair of samples $x_i$ and $x_j$, is expressed by:

$$K(x_i, x_j) = e^{-\gamma||x_i - x_j||^2} \tag{1}$$

Parameter `C` can be interpreted as the inverse of regularization. Parameter `gamma` ($\gamma$) controls the influence that the classification of a given training sample has over the classification of its neighbors, where a larger `gamma` means that only closer samples are affected. Natively, `SVC` only supports binary classification, but it was extended with a one-versus-one approach to allow multi-class classification. All attempts to classify the land cover with SVMs, were done with 5 classes and the scikit-learn `SVC` classifier, which is based on `LibSVM` [14] [15].

The second ML model evaluated was Random Forest (RF), a supervised Machine Learning algorithm based on the concept of ensemble learning [16]. An example of a successful application of a RF model to land cover classification is documented in [10]. RF improves the Decision Tree (DT) algorithm, and emerged with the objective of minimizing its main limitations: they are prone to overfitting and even a small change in the training data can result in a huge difference on the decision tree structure. The random forest overcomes these limitations by taking the prediction from each tree and based on the majority votes from the trees (figure 1). It uses bagging and feature randomness when building each individual tree, in order to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree. Randomness is built into RF mainly in two ways: each tree is fitted on a subset of the entire dataset, and each tree can grow differently, by virtue of the randomized order or subset of the features considered for optimum split in the Decision Tree.

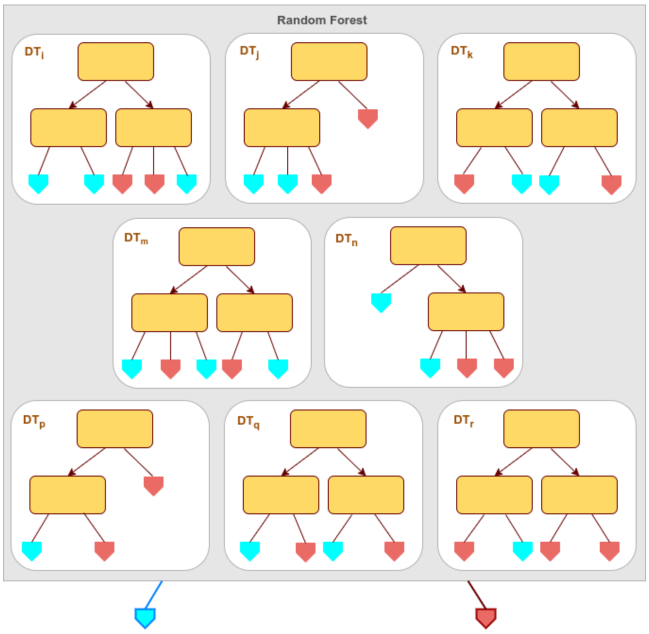


**Figure 1.** A RF model is a forest of decision trees.

The Random Forest uses an ensemble technique called Bootstrap Aggregating, or Bagging. First, each decision tree is trained independently with a different bootstrapped set, obtained from the entire dataset using sampling with replacement (bootstrap step). During inference, a prediction is made by each decision tree, and the final prediction by the random forest is returned as a majority vote (aggregation step). The cost function, or criterion, used more often during the learning process to split a node of the decision tree is called the Gini Impurity. It is basically a concept to quantify how homogeneous or "pure" a node is. A node is considered pure (G=0) if all training samples in the node belong to the same class, while a node with many training samples from many different classes will have a Gini Impurity close to 1. The Gini impurity at a node is computed by equation 2.

$$G = 1 - \sum_{c=1}^{NC} \frac{n_c}{n}^2 \tag{2}$$

Where $NC$ is the number of classes, $n_c$ is the number of samples belonging to class $c$ on the node, and $n$ is the total number of samples on the node.

In the present work RFs were implemented with the scikit-learn `RandomForestClassifier`. The most relevant hyperparameters of `RandomForestClassifier` are the number of trees the algorithm builds (`n_estimators`), the maximum number of features considered when splitting a node (`max_features`), and the minimum number of samples that must be allocated to each leaf node to be created (`min_sample_leaf`).

The last model, and the one that was most thoroughly evaluated, to classify the land cover was the neural network U-Net. U-Net is a CNN model initially developed for biomedical image segmentation and to be trained with few images [17]. However, both U-Net and other variants of it, were successfully applied to the RS domain, as reported in works [1] [7] [18] [19].

As can be seen in figure 2, U-Net comprises two parts, a contracting path that captures context (top part), and a symmetric expanding path that enables precise localization of features (bottom part). The contracting part extracts features through convolutions with 3 × 3 filters and max pooling layers. The expanding part uses convolutions and transposed convolutions to reduce the number of feature maps from 512 to 64, while it increases their dimensions from 15 × 15 to 120 × 120. Feature maps from the contracting part of the network are copied to the expanding part to avoid losing spatial information. The copy is implemented by the 4 vertical skip connections in figure 2. The copied features are then concatenated with same size features from the expanding path.

In our experiments, the U-Net receives as input 120 × 120 patches and outputs `#classes` segmentation masks with the same size, one mask per land cover class. As documented in the next section, experiments were carried out with different numbers of land cover classes.

## 4. Experiments and Results

### *4.1. Support Vector Machine Classifier*

In the first experiment with SVMs, the model was trained with unbalanced samples from 256 image patches of size 120x120 pixels and 10 bands. The model was trained with `C=2.0`, the RBF kernel, `gamma='scale'`, unlimited number of iterations, and `decision_function_shape='ovr'`. The achieved validation accuracy was 79.3%. Since the dataset is quite unbalanced, a reasonable high accuracy is achieved by a model that is tuned to classify correctly the 3 most frequent classes (1, 2, 5) and misclassifying the least frequent ones (0, 3). The next step was to balance the dataset, considering the same number of samples for all the classes. The considered number of samples was defined as the minimum value of the occurrences among the 5 classes.

Another direction that was explored was applying Principal Component Analysis (PCA) to reduce the number of features per sample from 10 (bands) to 3 (principal components), those that explain around 99% of the variance. Figure 3 shows the result of plotting the samples, after being projected on a 2D/3D space, defined by the two/three principal components of PCA that explain most of the variance. The projection on 3D makes it easier to visualize the clustering of the samples belonging to the same class. The visual analysis of this figure reveals that the classes exhibit a significant overlapping on the 3D space, which will make separation difficult. It was applied grid search cross-validation (CV) to find best values for the hyperparameters C and $gamma$ of the SVM model. It was found that $C = 1.0$ and $gamma = 5.0$ allow the best accuracy. When using a $C \geq 10000$ it was observed that the computation time, necessary to run a "batch" with a combination of hyperparameters,

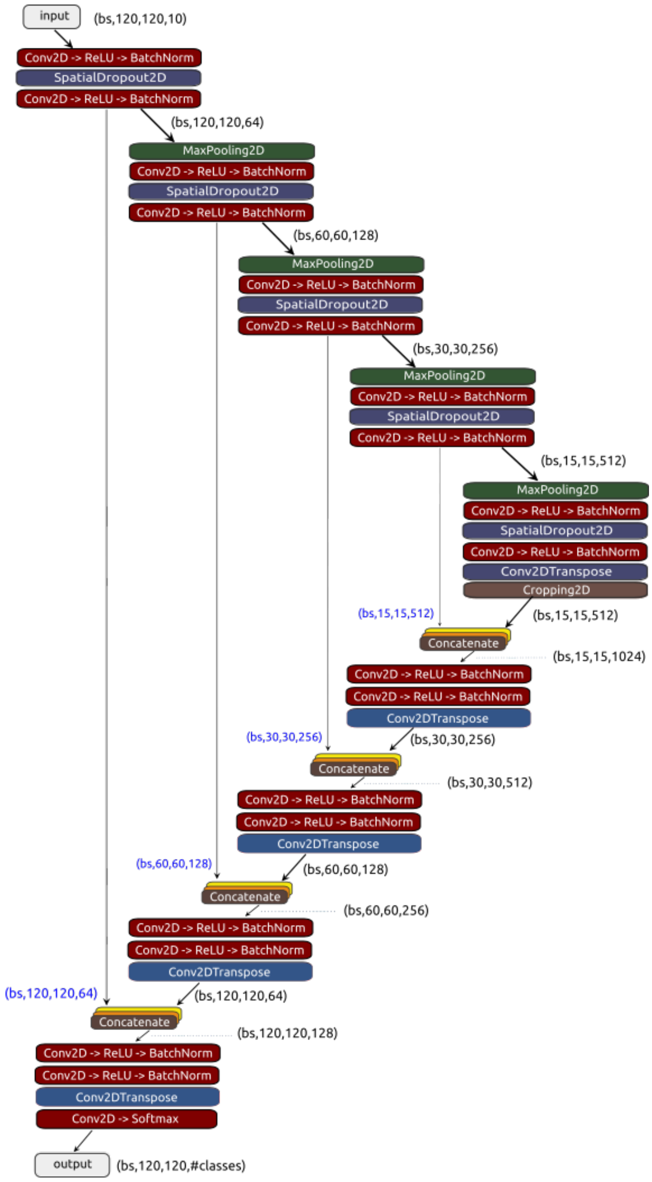


**Figure 2.** U-Net model trained on patches with 120x120 pixels and 10 bands.

became extremely high. The global test accuracy of the model was 59.1%. Finally, we dropped PCA and keep the 10 original features per pixel. Considering 2048 image patches, 10 features per pixel (corresponding to 10 Sentinel-2 bands) which allowed us to achieve the highest accuracy with SVM, 68.6%.

Evaluation metrics for the best SVM model are presented in table 1 and the confusion matrix is in figure 4. Considering the F1-score, the worst result belongs to the wetlands class (class 3). Although the improvement of the SVM model after balancing the dataset, optimizing the hyperparameters and reducing the number of features with PCA is not a satisfactory result and reveals that SVM is not the best fit to classify the land cover. Moreover, even a moderated number of image patches, such as 1024, turns the training very slow.

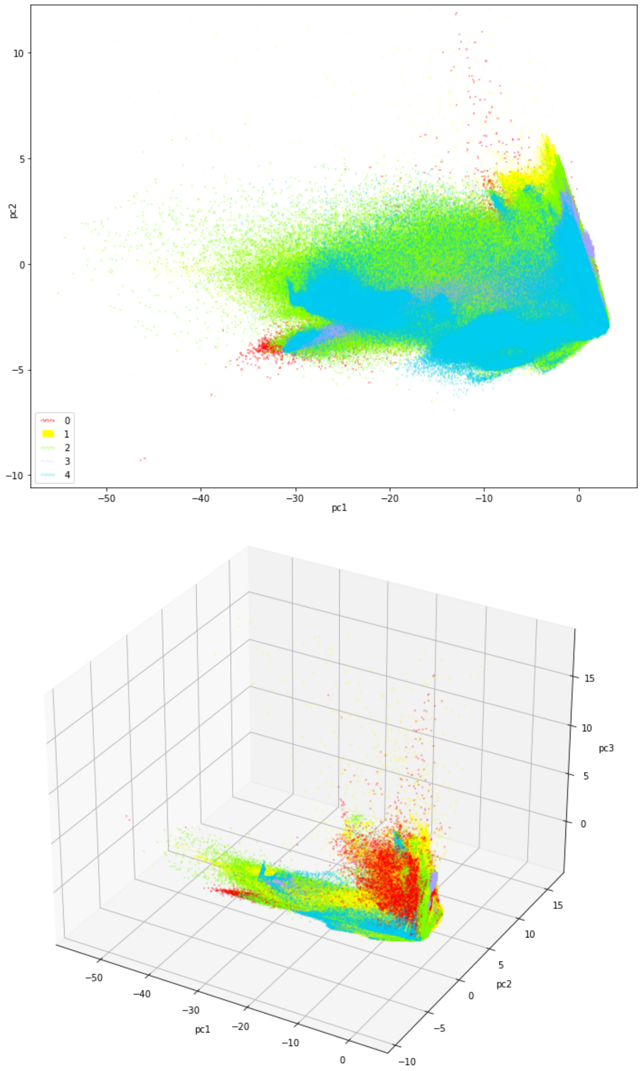


**Figure 3.** Plotting the samples after being projected on a 2D/3D space defined by the two/three principal components of PCA.

**Table 1.** Results for SVM model, using 10 features per sample, and the 5 CLC level 1 classes.

| Class | Precision | Recall | F1-score |
|---|---|---|---|
| 0 - Artificial surfaces | 0.69 | 0.65 | 0.67 |
| 1 - Agricultural areas | 0.57 | 0.67 | 0.62 |
| 2 - Forest and semi-natural areas | 0.59 | 0.73 | 0.66 |
| 3 - Wetlands | 0.75 | 0.46 | 0.57 |
| 4 - Water bodies | 0.89 | 0.91 | 0.90 |

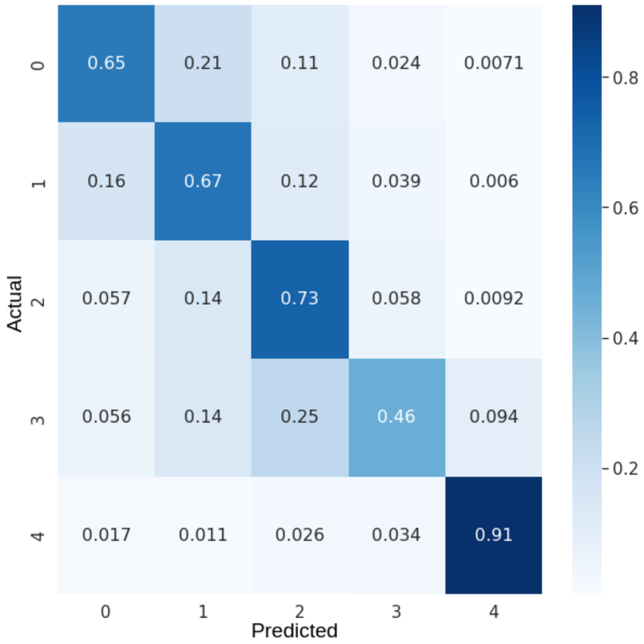


**Figure 4.** Normalized confusion matrix for the classification in 5 classes with SVM.

### *4.2. Random Forest Classifier*

Training of the RF was done with 1024 image patches of 120x120 pixels each, the number of land cover classes was 5, classes were balanced by considering a number of pixels per class equal to the least frequent class, PCA was applied to select the 3 features that explain most of the variance, the criterion used to evaluate the splits was `log_loss`, the RF included 100 decision trees, `bootstrap=False` meaning the whole dataset is applied to train each tree. The test accuracy score achieved by the trained model is 0.557.

Next, the number of image patches was increased to 2048, the number of features per pixel remained on 3, the evaluation criterion was changed to `gini`, the number of decision trees was kept on 100, `bootstrap=True` and `max_samples= 0.8`. The test accuracy score achieved by the trained model is 0.573. It was also tried increasing the number of decision trees to 200, but there was no improvement on the model performance.

Since using only 3 features per pixels resulted in poor results, it was decided to remove PCA and keep the 10 original features per pixel. Considering 2048 image patches, 10 features per pixel (corresponding to 10 Sentinel-2 bands), 5 land cover classes, balancing the frequency of the classes, with the `gini` evaluation criterion, `bootstrap=True`, `max_samples= 0.8`, and `max_features=3`, the test accuracy score achieved by the trained model raised to 0.706. The confusion matrix is presented in figure 5. This confusion matrix reveals that the percentage of samples correctly classified is 68.0% for class 0, 67.0% for class 1, 72.0% for class 2, 55.0% for class 3, and 93.0% for class 4. Precision, recall, and F-score metrics for the trained RF model are shown on table 2.

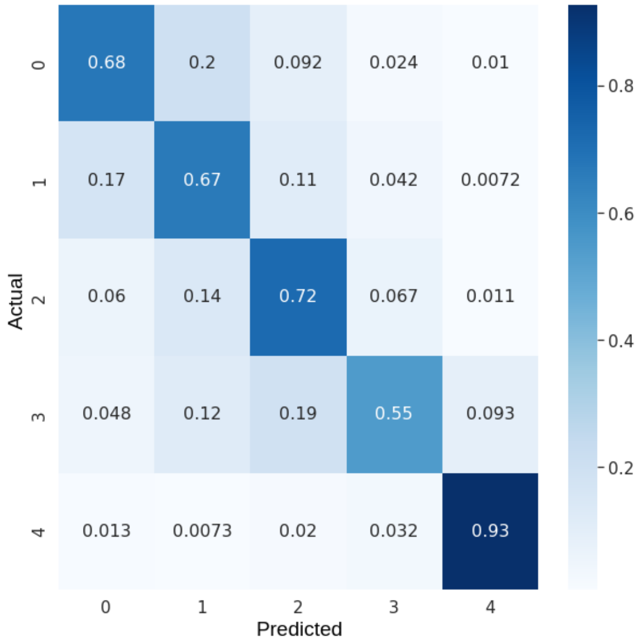


**Figure 5.** Normalized confusion matrix for the classification in 5 classes with RF.

**Table 2.** Results for RF model, using 10 features per sample, and the 5 CLC level 1 classes.

| Class | Precision | Recall | F1-score |
|---|---|---|---|
| 0 - Artificial surfaces | 0.70 | 0.68 | 0.69 |
| 1 - Agricultural areas | 0.59 | 0.67 | 0.62 |
| 2 - Forest and semi-natural areas | 0.63 | 0.72 | 0.67 |
| 3 - Wetlands | 0.77 | 0.55 | 0.64 |
| 4 - Water bodies | 0.88 | 0.93 | 0.91 |

### 4.3. *U-Net*

U-Net model was described with the TensorFlow library, especially the Keras API, it was trained with Adam optimizer, the categorical cross-entropy loss, the `ModelCheckpoint`, `EarlyStopping`, and `ReduceLROnPlateau` cllbacks, during 200 epochs. Models were evaluated based on accuracy, precision, recall, and F1-score metrics.

A list of all experiments carried out, as well as the results obtained, can be seen in the table 3. The set of experiments accomplished with U-Net and the BigEarthNet dataset will be summarized now.

**Table 3.** Summary of the different experiments.

| Model | Classes | Dataset | Overall Accuracy |
|---|---|---|---|
| SVM | 5 | BigEarthNet | 68.6% |
| RF | 5 | BigEarthNet | 70.6% |
| U-Net | 43 | BigEarthNet | 82.32% |
| U-Net + NDVI | 43 | BigEarthNet | 77.95% |
| U-Net | 15 | BigEarthNet | 87.11% |
| U-Net | 11 | BigEarthNet | 86.88% |
| U-Net | 8 | BigEarthNet | 92.37% |
| U-Net | 5 | BigEarthNet | 94.75% |
| U-Net | 5 | LandCoverPT | 87.26% |

The experiment with all 43 CLC level 3 classes will work as our baseline, i.e, with all the other experiments we will try to improve the results of the baseline. The overall accuracy achieved was 82.32% with most misclassifications being within very similar classes, such as continuous urban fabric and discontinuous urban fabric. The class with the lowest results was green urban areas, being misclassified as urban fabric or forests.

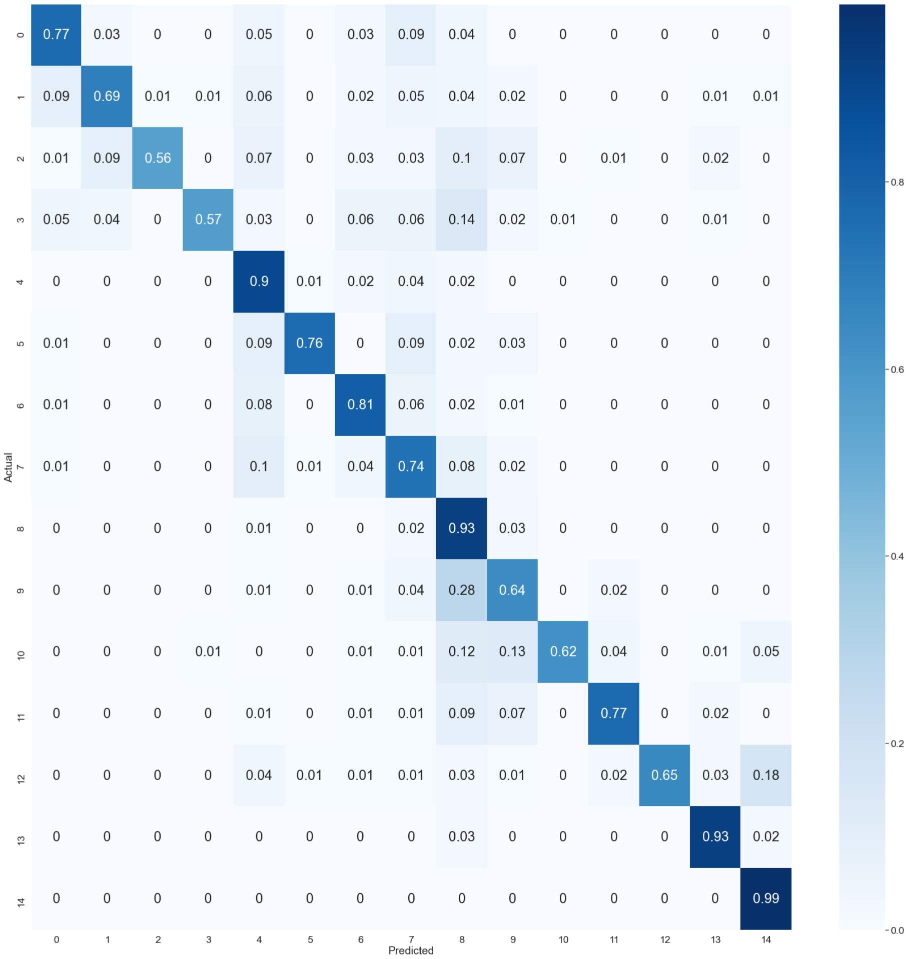
The second experiment tried to improve the results of the previous attempt through the insertion of the Normalized Difference Vegetation Index (NDVI). NDVI was chosen because of its popularity in the literature, for example in [20] and [21]. The final results were worse than in the previous scenario, analysing each class individually shows that some classes were being completely misclassified and this did not happen in the previous experiment. Taking into consideration these results the idea of using other spectral indexes was abandoned.

The next step taken to improve the results was to reduce the number of land cover classes. The experiment with 15 CLC level 2 classes improved the overall accuracy to 87.11%. The normalized confusion matrix for the segmentation in 15 classes with U-Net is shown in figure 6. While the baseline presented some values for the F1-score metric of the order of 0.4, this model presents 0.65 as the lowest value.

The automatic classification of land cover in 15 classes is still a very ambitious objective, and therefore another model was trained to classify the land cover only in the 5 CLC level 1 classes. The trained U-Net model achieved a 94.75% overall accuracy, the best result among all experiments. Evaluation metrics for this model are presented in table 4 an the confusion matrix is in figure 7. Considering the F1-score, the worst result belongs to the wetlands class (class 3).

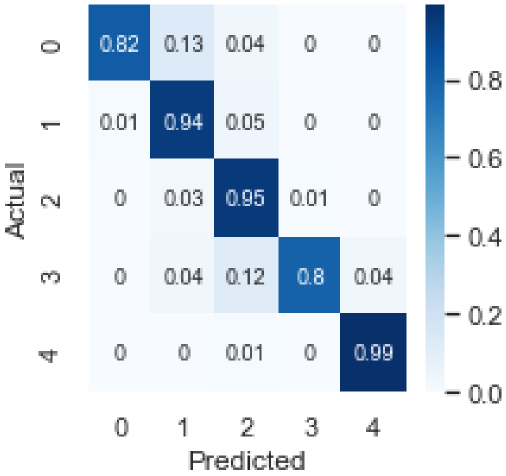
**Table 4.** Results for U-Net model, using 10 spectral bands and the 5 CLC level 1 classes.

| **Class** | **Precision** | **Recall** | **F1-score** |
|---|---|---|---|
| 0 - Artificial surfaces | 0.86 | 0.82 | 0.84 |
| 1 - Agricultural areas | 0.94 | 0.94 | 0.94 |
| 2 - Forest and semi-natural areas | 0.95 | 0.95 | 0.95 |
| 3 - Wetlands | 0.77 | 0.80 | 0.78 |
| 4 - Water bodies | 0.98 | 0.99 | 0.98 |



| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0.77 | 0.03 | 0 | 0 | 0.05 | 0 | 0.03 | 0.09 | 0.04 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0.09 | 0.69 | 0.01 | 0.01 | 0.06 | 0 | 0.02 | 0.05 | 0.04 | 0.02 | 0 | 0 | 0 | 0.01 | 0.01 |
| 2 | 0.01 | 0.09 | 0.56 | 0 | 0.07 | 0 | 0.03 | 0.03 | 0.1 | 0.07 | 0 | 0.01 | 0 | 0.02 | 0 |
| 3 | 0.05 | 0.04 | 0 | 0.57 | 0.03 | 0 | 0.06 | 0.06 | 0.14 | 0.02 | 0.01 | 0 | 0 | 0.01 | 0 |
| 4 | 0 | 0 | 0 | 0 | 0.9 | 0.01 | 0.02 | 0.04 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0.01 | 0 | 0 | 0 | 0.09 | 0.76 | 0 | 0.09 | 0.02 | 0.03 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0.01 | 0 | 0 | 0 | 0.08 | 0 | 0.81 | 0.06 | 0.02 | 0.01 | 0 | 0 | 0 | 0 | 0 |
| 7 | 0.01 | 0 | 0 | 0 | 0.1 | 0.01 | 0.04 | 0.74 | 0.08 | 0.02 | 0 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0.02 | 0.93 | 0.03 | 0 | 0 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0.01 | 0.04 | 0.28 | 0.64 | 0 | 0.02 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0.01 | 0.01 | 0.12 | 0.13 | 0.62 | 0.04 | 0 | 0.01 | 0.05 |
| 11 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0.01 | 0.01 | 0.09 | 0.07 | 0 | 0.77 | 0 | 0.02 | 0 |
| 12 | 0 | 0 | 0 | 0 | 0.04 | 0.01 | 0.01 | 0.01 | 0.03 | 0.01 | 0 | 0.02 | 0.65 | 0.03 | 0.18 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.03 | 0 | 0 | 0 | 0 | 0.93 | 0.02 |
| 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.99 |

**Figure 6.** Normalized confusion matrix for the segmentation in 15 classes with U-Net.



| | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 0 | 0.82 | 0.13 | 0.04 | 0 | 0 |
| 1 | 0.01 | 0.94 | 0.05 | 0 | 0 |
| 2 | 0 | 0.03 | 0.95 | 0.01 | 0 |
| 3 | 0 | 0.04 | 0.12 | 0.8 | 0.04 |
| 4 | 0 | 0 | 0.01 | 0 | 0.99 |

**Figure 7.** Normalized confusion matrix for the segmentation in 5 classes with U-Net.

Two attempts with a combination of CLC classes from levels 1 and 2 were realized. The first one used 11 classes and obtained an overall accuracy of 86.11%, a result worse than the experiment with 15 classes.

The second attempt used 8 classes and its overall accuracy was 92.37%, a result very similar to the experiment with 5 classes (table 5). The chosen level 2 classes are those that

doi:10.20944/preprints202302.0102.v1

were best classified by U-Net trained with the level 2 classes. The remaining level 2 classes were collapsed into the corresponding level 1 classes. The CLC hierarchy was maintained, i.e, only level 2 classes that would be part of the same level 1 class were gathered. Confusion matrix analysis (figure 8) shows that 11% of the samples belonging to class 0 (artificial surfaces) are classified as class 1 (agricultural areas), 12% of class 2 (pastures) is classified as class 1 (agricultural areas), 17% of class 4 (inland wetlands) is classified as class 3 (forest and semi-natural areas), and 18% of class 5 (maritime wetlands) is classified as class 7 (maritime waters).

**Table 5.** Results for U-Net model, using 10 spectral bands and 8 CLC level 1 and level 2 classes.

| Class | Precision | Recall | F1-score |
|---|---|---|---|
| 0 - Artificial surfaces | 0.84 | 0.81 | 0.83 |
| 1 - Agricultural areas | 0.91 | 0.88 | 0.90 |
| 2 - Pastures | 0.81 | 0.83 | 0.82 |
| 3 - Forest and semi-natural areas | 0.94 | 0.96 | 0.95 |
| 4 - Inland wetlands | 0.78 | 0.76 | 0.77 |
| 5 - Maritime wetlands | 0.79 | 0.72 | 0.76 |
| 6 - Inland waters | 0.94 | 0.94 | 0.94 |
| 7 - Maritime waters | 0.99 | 0.99 | 0.99 |

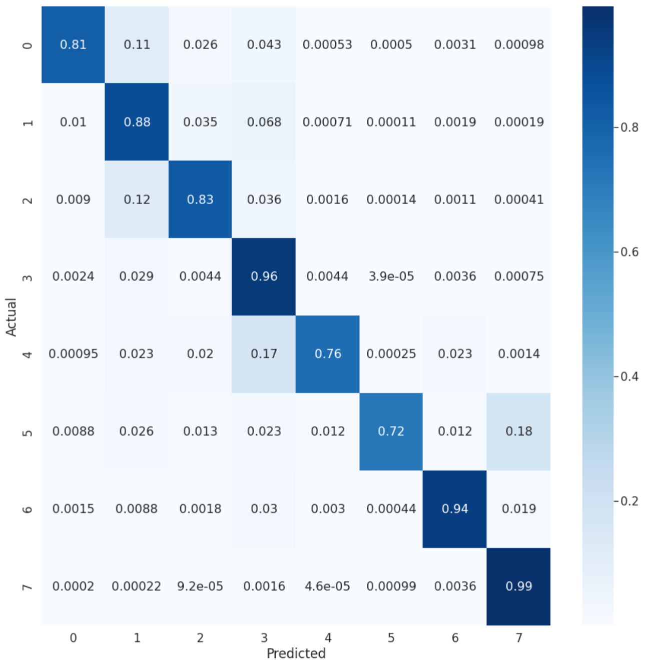


**Figure 8.** Normalized confusion matrix for the segmentation in 8 classes with U-Net.

Experiments with the LandCoverPT dataset, the U-Net model, and level 1 land cover classes, were also accomplished. The results of these experiments were worse than those achieved with the BigEarthNet dataset, quantified as an overall accuracy of 87.26%. Classes with the worst results in this experiment were artificial surfaces and wetlands. A possible explanation for this results can be the low number of samples containing those classes in the LandCoverPT dataset.

The visual inspection to the predictions with the trained models, and to the correspondent ground-truth, revealed that the classification errors were predominantly located at the

doi:10.20944/preprints202302.0102.v1

boundary of the patches (figure 9). The most likely explanation for this fact is the existence of mixed pixels. Another explanation, mentioned in the literature, is the lower ability of U-Net to correctly segment pixels at the object's boundaries.

Figure 9 shows a satellite image patch, randomly chosen from the test set. The leftmost column of the figure shows the ground truth masks for the 5 level 1 classes (0 to 4). The next column shows the model prediction for the same classes. In the upper right corner are presented 4 of the 10 bands of the input patch, in this case the ones with the best spatial resolution: red, green, blue and near infrared. Pixels that were misclassified are shown in yellow in the central right part of the figure.

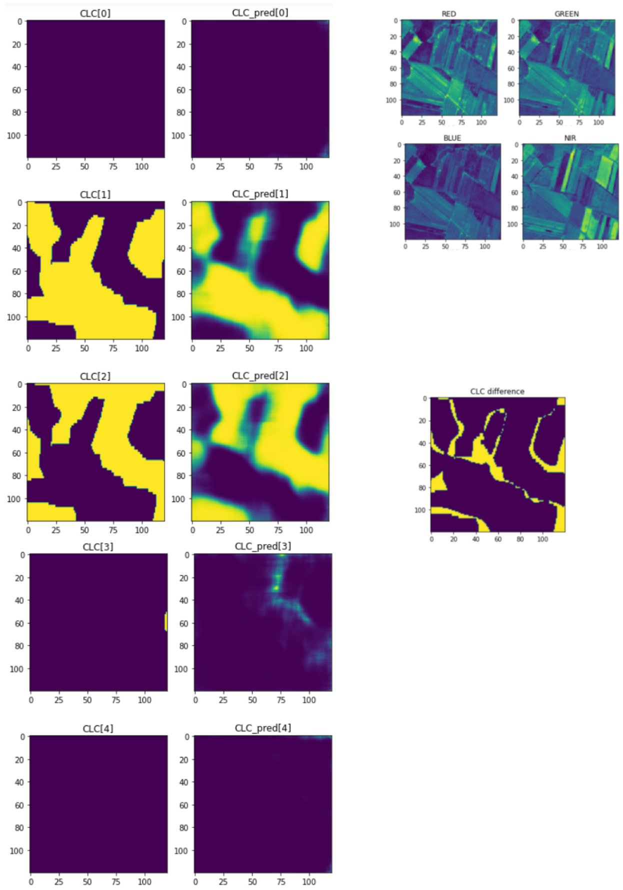


**Figure 9.** Random satellite image patch from the test set: input masks (left), predicted masks (middle), input visible bands (top right), and misclassified pixels (center right).

When we evaluate the trained models with a dataset distinct from the train/validation set, the results are inferior. It was observed that several land cover parcels, classified as agricultural areas in the 2018 CLC map (yellow regions in figure 10), are misclassified by our models as artificial surfaces (red regions in figure 10) or forests and semi-natural areas (green regions in figure 10).

Another problem, observed in some parts of the automatically generated map, is the discontinuity between patches. This problem occurs because the masks generated by the model are obtained patch by patch, where the patch size is 120 × 120. A possible solution is to discard the pixels on the periphery of the patches and use only the inner part (figure 11). The innermost pixels have more contextual information and better accuracy than peripheric pixels, as it can be seen in table 6. The drawback of this solution is the longer time is takes to generate the land cover map. For example, considering only a inner part of 20 × 20 pixels on each patch, the time to classify the same land area will increase $6 * 6$ times.

Figure 12 contains a complete and continuous land cover map for continental Portugal. This map was generated with the U-Net model, trained on the BigEarthNet dataset and 5

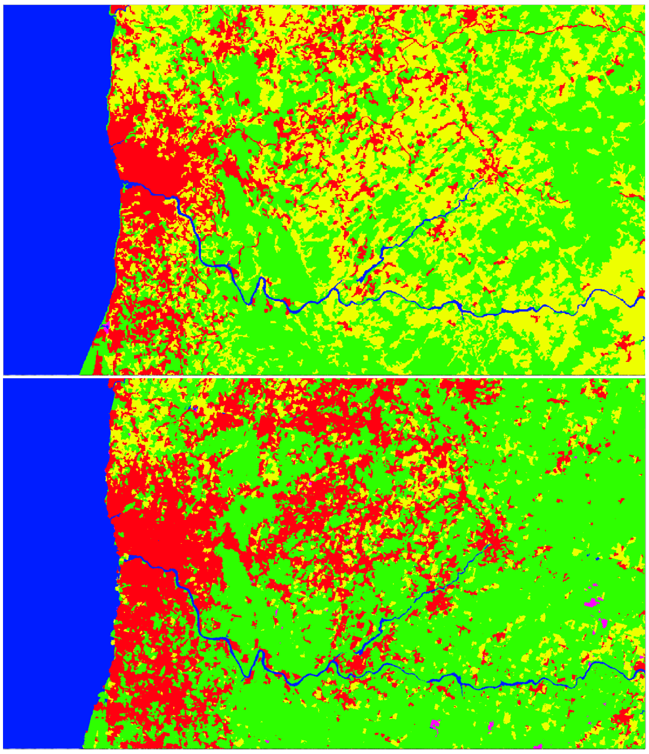


**Figure 10.** Ground truth 2018 CLC map with 5 classes, for the northwest region of Portugal (top) and corresponding map generated by the trained U-Net model (bottom). Color scheme: red - artificial surfaces, yellow - agricultural areas, green - forest and semi-natural areas, magenta - wetlands, blue - water bodies.

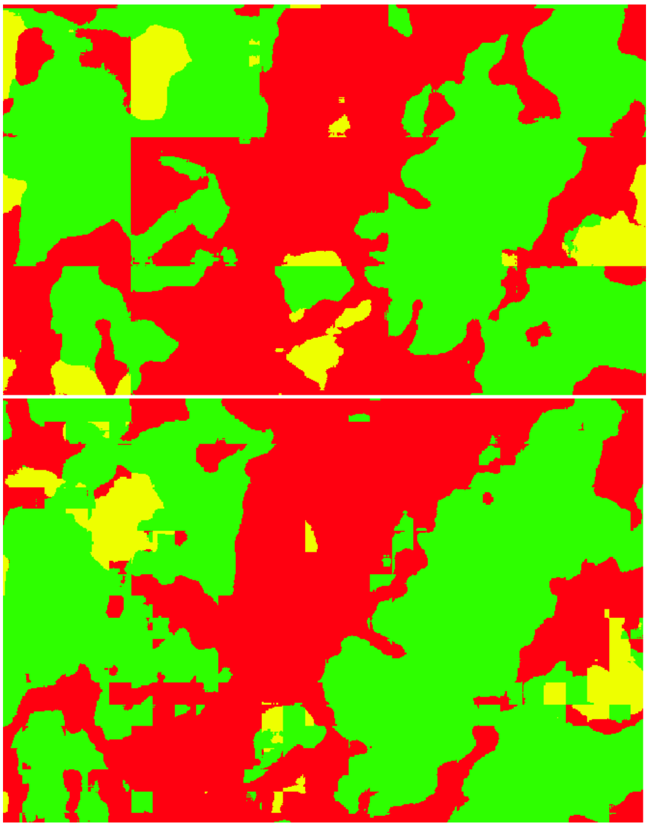


**Figure 11.** Discontinuity problem in the segmentation of patches (top) and its mitigation (bottom).

**Table 6.** Accuracy achieved when using only the inner part of each patch, for different sizes of the considered area.

| Size of the area used on each patch | 43 CLC classes | 15 CLC classes | 5 CLC classes |
|---|---|---|---|
| 120 × 120 | 82.38% | 87.02% | 94.75% |
| 110 × 110 | 82.71% | 87.27% | 94.87% |
| 100 × 100 | 82.91% | 87.42% | 94.94% |
| 90 × 90 | 83.07% | 87.51% | 95.00% |
| 80 × 80 | 83.17% | 87.60% | 95.03% |
| 70 × 70 | 83.26% | 87.68% | 95.08% |
| 60 × 60 | 83.37% | 87.74% | 95.08% |
| 50 × 50 | 83.43% | 87.76% | 95.08% |
| 40 × 40 | 83.50% | 87.83% | 95.09% |
| 30 × 30 | 83.55% | 87.88% | 95.11% |
| 20 × 20 | 83.59% | 87.89% | 95.11% |
| 10 × 10 | 83.62% | 87.93% | 95.10% |

classes. Sentinel-2 products, downloaded from the https://scihub.copernicus.eu website, were used to generate the full map. Images were captured by the satellite on July 7, 2021 and August 22, 2021, and have a maximum cloud percentage of 5%. Because products with a minimum cloud percentage were needed, it was impossible to use all the images from the same day. To visualize the map we used the QGIS tool, where the various parcels of the map generated by the model were merged and trimmed with the help of a shapefile that defines the boundaries of the Portuguese mainland.

## 5. Conclusions and Future Work

The results achieved in the present work provide an evidence that it is possible to automatically and reliably generate an updated land cover map. Thus, the results of this study are relevant for those working in the field of remote sensing.

The biggest difficulty encountered in the course of the work was the processing of large amounts of data from a dataset such as the BigEarthNet or the Sentinel-2 satellite products. To overcome these difficulties techniques such as feeding the training loop with data stored in TFrecords files and adopting iterative processes whenever possible.

The best trained model achieved an overall accuracy of 94.75%, which can be increased to 95.11% if only the central pixels of the patches are considered during the segmentation of each patch. Although this result is very good, it should however be taken into consideration that the visual comparison between the official 2018 CLC map and the map generated by the developed model, for the same geographical area and the same year, shows that the overall quality of the generated map is lower than 94.75%.

When classifying land cover into 5 classes, a consistent result across all models is a greater difficulty in identifying artificial surfaces (class 0) and wetlands (class 4). The explanation lies in the similarity between the spectral characteristics of artificial surfaces and agricultural areas (class 1), and between wetlands and semi-natural areas (class 2). In the case of Portuguese territory, the identification of class 3 constitutes an added problem because wetlands are not frequent.

The latest official CLC map is relative to 2018 and required a production time of about one and a half year. While training, tuning and generating the land cover map with the proposed model requires a time of less than a month. The ML model will never have a higher accuracy than the CLC project since the model learns from the official CLC map

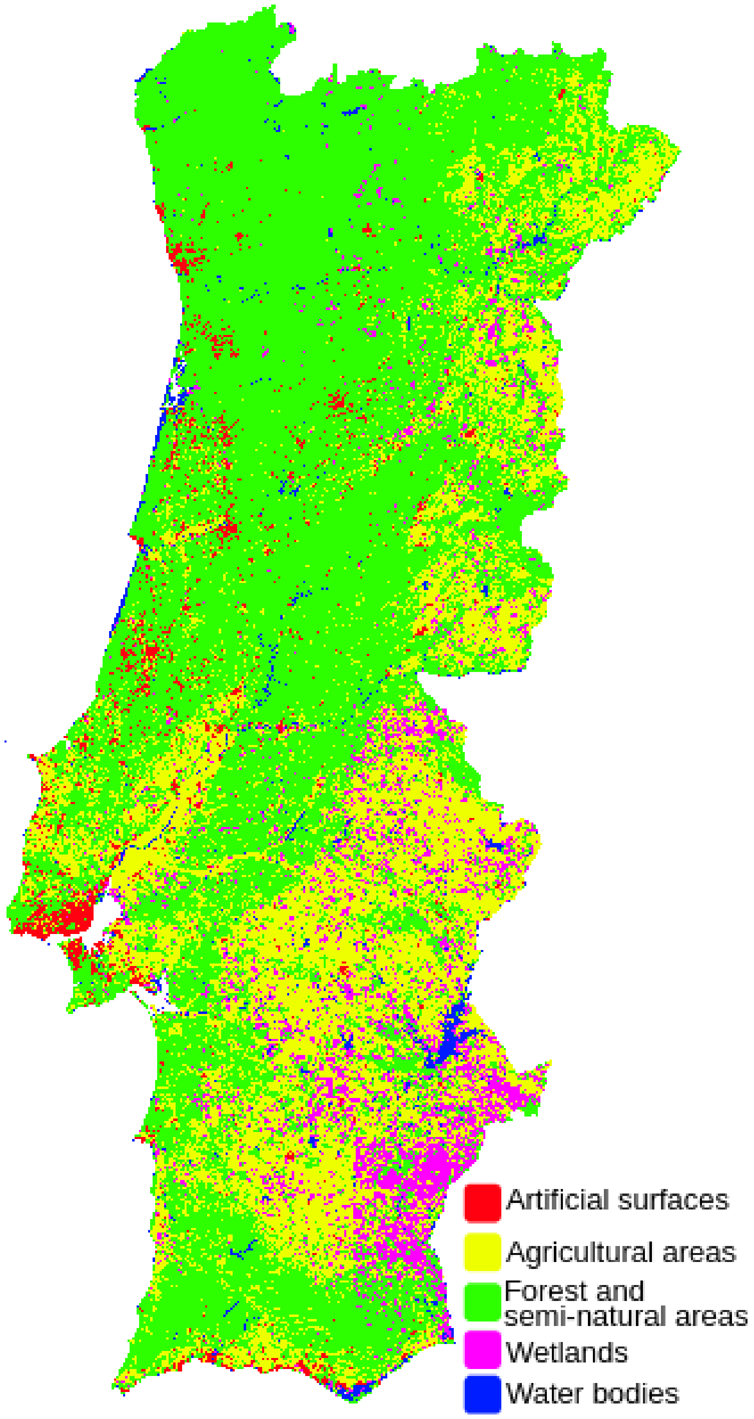


**Figure 12.** Land cover map for the Portuguese mainland generated by the U-Net model.

data. However, given the time difference needed to produce the maps, the maps generated with ML models are feasible in several scenarios, because they may be more up-to-date than the official CLC map.

To conclude it is necessary to point out that the CLC maps and the Land Use and Land Cover charts have a human error, and when these maps are used to train ML models the error remains. In case of the 2018 CLC map, each participating country commissioned a team to create their map, but all countries used the same methodology and nomenclature, to ensure an accuracy higher than 85%.

Although the best model achieved good results, some alternatives remained to be explored. Here are some possibilities to improve the presented results:

- Test other segmentation models that address some of the U-Net limitations, such as models based on Feature Pyramid Networks [22] [23] [24] [25] [26] and DeepLab [27].
- Test other datasets, improve and increase the tested LandCoverPT dataset, which exhibit some limitations to obtain optimal results. Another possibility is to improve the dataset would be to optimize the size of the patches into which the Sentinel-2 products were divided.
- Implement other strategies to minimize the segmentation problem at the periphery of patches.
- Take a more consistent approach to optimizing model hyperparameters, for example by using a library such as Optuna or TPOT.
- Add other types of data to the optical images, such as radar images collected by the Sentinel-1 satellite.
- Test spectral indexes with the random forest model.

**Author Contributions:** Conceptualization, A.E.; methodology, A.E. and N.V.; software, A.E. and N.V.; validation, A.E. and N.V.; investigation, A.E. and N.V.; resources, N.V.; data curation, A.E.; writing—original draft preparation, N.V.; writing—review and editing, A.E.; visualization, A.E. and N.V.; supervision, A.E.; project administration, A.E.; funding acquisition, A.E. All authors have read and agreed to the published version of the manuscript.

**Data Availability Statement:** Data used in the present work can be found on `https://bigearth.net` (the BigEarthNet dataset) and `https://www.kaggle.com/datasets/ajesteves/landcoverpt` (the LandCoverPT dataset).

**Acknowledgments:** This work has been supported by FCT – Fundação para a Ciência e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020.

**Conflicts of Interest:** The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

## References

1. Mäyrä, J. Land cover classification from multispectral data using convolutional autoencoder networks. Master's thesis, University of Jyväskylä, 2018.
2. Sumbul, G.; Charfuelan, M.; Demir, B.; Markl, V. Bigearthnet: A Large-Scale Benchmark Archive for Remote Sensing Image Understanding. In Proceedings of the IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, 2019, pp. 5901–5904. https://doi.org/10.1109/IGARSS.2019.8900532.
3. Ma, L.; Li, M.; Ma, X.; Cheng, L.; Du, P.; Liu, Y. A review of supervised object-based land-cover image classification. *ISPRS Journal of Photogrammetry and Remote Sensing* **2017**, *130*, 277 – 293. https://doi.org/https://doi.org/10.1016/j.isprsjprs.2017.06.001.
4. Syrris, V.; Hasenohr, P.; Delipetrev, B.; Kotsev, A.; Kempeneers, P.; Soille, P. Evaluation of the Potential of Convolutional Neural Networks and Random Forests for Multi-Class Segmentation of Sentinel-2 Imagery. *Remote Sensing* **2019**, *11*, 907. https://doi.org/10.3390/rs11080907.
5. Georganos, S.; Grippa, T.; Lennert, M.; Vanhuysse, S.; Johnson, B.; Wolff, E. Scale Matters: Spatially Partitioned Unsupervised Segmentation Parameter Optimization for Large and Heterogeneous Satellite Images. *Remote Sensing* **2018**, *10*, 1440. https://doi.org/10.3390/rs10091440.
6. Zhang, C. Deep Learning for Land Cover and Land Use Classification. PhD thesis, Lancaster University, 2018. https://doi.org/10.17635/lancaster/thesis/428.
7. Zhang, X.; Han, L.; Han, L.; Zhu, L. How Well Do Deep Learning-Based Methods for Land Cover Classification and Object Detection Perform on High Resolution Remote Sensing Imagery? *Remote Sensing* **2020**, *12*.
8. Liu, D.; Xia, F. Assessing object-based classification: advantages and limitations. *Remote Sensing Letters* **2010**, *1*, 187–194, [https://doi.org/10.1080/01431161003743173]. https://doi.org/10.1080/01431161003743173.

9. Abdi, A.M. Land cover and land use classification performance of machine learning algorithms in a boreal landscape using Sentinel-2 data. *GIScience & Remote Sensing* **2020**, *57*, 1–20, [https://doi.org/10.1080/15481603.2019.1650447]. https://doi.org/10.1080/15481603.2019.1650447.
10. Van Tricht, K.; Gobin, A.; Gilliams, S.; Piccard, I. Synergistic Use of Radar Sentinel-1 and Optical Sentinel-2 Imagery for Crop Mapping: A Case Study for Belgium. *Remote Sensing* **2018**, *10*, 1642. https://doi.org/10.3390/rs10101642.
11. Sumbul, G.; Charfuelan, M.; Demir, B.; Markl, V. Bigearthnet: A Large-Scale Benchmark Archive for Remote Sensing Image Understanding. In Proceedings of the IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, 2019, pp. 5901–5904. https://doi.org/10.1109/IGARSS.2019.8900532.
12. Boser, B.; Guyon, I.; Vapnik, V. A training algorithm for optimal margin classifiers. In Proceedings of the Proceedings of the 5th Workshop on Computational Learning Theory, 1992, pp. 144—-152. https://doi.org/10.1145/130385.130401.
13. Cortes, C.; Vapnik, V. Support-vector networks. *Machine Learning* **1995**, *20*, 273—297. https://doi.org/10.1007/BF00994018.
14. Chang, C.C.; Lin, C.J. LIBSVM: A Library for Support Vector Machines, 2001.
15. Hsu, C.W.; Chang, C.C.; Lin, C.J. A Practical Guide to Support Vector Classification, 2016.
16. Breiman, L. Random Forests. *Machine Learning* **2001**, *45*, 5–32. https://doi.org/10.1023/-A:1010933404324.
17. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the Int. Conf. on Medical Image Computing and Computer-assisted Intervention, 2015, p. 234–241.
18. Stoian, A.; Poulain, V.; Inglada, J.; Poughon, V.; Derksen, D. Land Cover Maps Production with High Resolution Satellite Image Time Series and Convolutional Neural Networks: Adaptations and Limits for Operational Systems. *Remote Sensing* **2019**, *11*, 1986. https://doi.org/10.3390/rs11171986.
19. Zhang, Z.; Liu, Q.; Wang, Y. Road Extraction by Deep Residual U-Net. *CoRR* **2017**, *abs/1711.10684*.
20. Gašparović, M.; Zrinjski, M.; Gudelj, M. Automatic cost-effective method for land cover classification (ALCC). *Computers, Environment and Urban Systems* **2019**, *76*, 1 – 10. https://doi.org/https://doi.org/10.1016/j.compenvurbsys.2019.03.001.
21. Clerici, N.; Calderón, C.A.V.; Posada, J.M. Fusion of Sentinel-1A and Sentinel-2A data for land cover mapping: a case study in the lower Magdalena region, Colombia. *Journal of Maps* **2017**, *13*, 718–726, [https://doi.org/10.1080/17445647.2017.1372316]. https://doi.org/10.1080/17445647.2017.1372316.
22. Seferbekov, S.S.; Iglovikov, V.I.; Buslaev, A.V.; Shvets, A.A. Feature Pyramid Network for Multi-Class Land Segmentation. *CoRR* **2018**, *abs/1806.03510*, [1806.03510].
23. Yuan, Z.; Liu, Z.; Zhu, C.; Qi, J.; Zhao, D. Object Detection in Remote Sensing Images via Multi-Feature Pyramid Network with Receptive Field Block. *Remote Sensing* **2021**, *13*. https://doi.org/10.3390/rs13050862.
24. Yuan, Y.; Fang, J.; Lu, X.; Feng, Y. Spatial Structure Preserving Feature Pyramid Network for Semantic Image Segmentation. *ACM Trans. Multimedia Comput. Commun. Appl.* **2019**, *15*. https://doi.org/10.1145/3321512.
25. Kirillov, A.; Wu, Y.; He, K.; Girshick, R.B. PointRend: Image Segmentation as Rendering. *CoRR* **2019**, *abs/1912.08193*, [1912.08193].
26. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid Scene Parsing Network. *CoRR* **2016**, *abs/1612.01105*, [1612.01105].
27. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In Proceedings of the ECCV, 2018.