NATIONAL UNIVERSITY OF SCIENCE AND TECHNOLOGY POLITEHNICA BUCHAREST

Doctoral School of Electronics, Telecommunications and Information Technology

Decision No. 82 from 19.07.2024

Ph.D. THESIS SUMMARY

Eng. Alexandru-Toma ANDREI

INTELLIGENT METHODS FOR IDENTIFYING TERRESTRIAL DETAILS BASED ON THE ANALYSIS OF MULTISPECTRAL AERIAL IMAGES

THESIS COMMITTEE

Prof. Dr. Ing. Ion MARGHESCU National Univ. Of Science and Technology POLITEHNICA Bucharest	President
Prof. Dr. Ing. Ovidiu GRIGORE National Univ. Of Science and Technology POLITEHNICA Bucharest	PhD Supervisor
Prof. Dr. Ing. Radu Gabriel BOZOMITU „Gheorghe Asachi" Technical Univ. of Iași	Referee
Prof. Dr. Ing. Radu-Viorel RĂDESCU National Univ. Of Science and Technology POLITEHNICA Bucharest	Referee
Conf. Dr. Ing. Alin DĂNIȘOR Constanța Maritime Univ.	Referee

BUCHAREST 2024

Content

Chapter 1

Introduction

1.1 Presentation of the field of the doctoral thesis

Historically, deforestation began long ago in the Holocene epoch [6]. Since humans discovered fire, they initiated a slow but steady process of global deforestation. Initially, wood was used for basic needs such as heating caves or cooking meat after hunting. These activities did not pose a problem for the forests, which could regenerate after tree cutting. However, humans began using wood to build houses, entire cities, or medieval walls, processing wood with water-powered mechanical saws installed along rivers. By the end of the medieval period, the forest was affected by two other elements: the invention of the steam engine and the increasing need for agriculture [7]. It is estimated that nearly half of the Earth's original forest has been lost [8]. In just the last century, 10% of the world's forests have been cleared for crops or pasture, with the peak occurring in the 1980s. In that decade, the total area deforested was equal to the size of present-day Mongolia [9].

In recent years, various organizations and activists have begun to fight to save the forest and stop uncontrolled deforestation. Society has already started to realize the importance of forests and the toxic effects of deforestation. International associations such as the European Union or non-governmental organizations regulate this activity to achieve an optimal balance between tree cutting and forest regeneration. But old habits die hard, and abuses or defiance of laws are still practiced. In addition to the entire known and legislated timber industry, illegal logging is another major problem. By applying the intelligent methods researched and developed in this work for forest identification, substantial assistance is provided to environmental activists and efforts to combat deforestation.

1.2 Scope of the doctoral thesis

This research aims to find a globally applicable method with low implementation and usage costs for detecting deforestation. The ultimate goal is for the final automated segmentation algorithm to be an easy-to-use tool for any activist or operator to identify deforestation. It is crucial to achieve quick results in addressing deforestation, considering modern capabilities in tree cutting, transportation, and processing. Thus, the objective boils down to identifying forests in multispectral aerial images, with examples and analyses conducted for the forest class.

In addition to these goals, a common impediment to artificial intelligence (AI) models is the need for resources. Most real-world implementations of AI models are carried out under challenging conditions or with very limited hardware capabilities, not to mention the need for fast data analysis and prediction. Another major objective

of this work was cost reduction. In the case of deforestation monitoring, this is a mandatory objective. Unfortunately, in the case of illegal logging, vast areas of forests can disappear overnight, so the method of combating it must be swift. Additionally, this method should be environmentally friendly and operate with minimal energy resources. Another issue with deforestation is the enforcement of laws or regulations. At present, the only method of detecting deforestation in real-time is through the patrols of volunteer activists in areas of interest and their reporting to the relevant authorities in case of suspicious activity. Given all of the above, the use of large aircraft for data collection is impractical. To achieve optimal results, monitoring of areas of interest must be carried out with small aerial platforms, such as drones, which can fly over areas of interest and report timely to a command center or relevant authorities. This would also address the issue of apprehending offenders or at least recording evidence against them.

Chapter 2

The current state of intelligent methods applied to multispectral images

In recent years, there has been an increasing demand for image interpretation, accompanied by advancements in machine learning techniques. Currently, most image classification works focus on Convolutional Neural Networks (CNNs) applied to already labeled open-source data. Most augmentation or improvement techniques are related to training the neural network and fine-tuning its parameters, while clustering methods have decreased in popularity. However, each type of machine learning has its advantages and disadvantages. For instance, supervised learning may be more accurate but requires labeled data, which can be resource-intensive. On the other hand, unsupervised learning does not require labeled data but can be challenging to find similarities or correlations among pixels in an aerial image and generally has lower accuracy. To better understand these characteristics, it was necessary to analyze the current state and related works to the doctoral thesis for each of the two approaches.

2.1 The current state of clustering techniques

Even though there are other methods such as Support Vector Machines (SVMs) or autoencoders, unsupervised learning is predominantly represented by clustering

techniques, as the goal is often to identify groups with similar features in a dataset, as is the case with ours. Clustering methods vary from the simplest to the most complex. The most common and straightforward clustering technique to implement is the K-Means algorithm. It is usually time-efficient and yields generally good results as it can adapt to most types of data. Often, this algorithm is implemented through various augmentation or preprocessing techniques tailored to the specific characteristics of each study [10-11].

Another clustering technique whose mathematical concepts can be traced back to the contributions of Carl Friedrich Gauss in the early 1800s is the Gaussian Mixture Model (GMM) algorithm. Although Gauss is renowned for developing the normal distribution, which serves as the fundamental basis for GMM clustering, the first explicit mention appears in the work of Karl Pearson in 1894 [13]. The method began to gain widespread recognition in the field of machine learning in the 1970s. A significant factor contributing to their popularity was the emergence of the Expectation Maximization (EM) algorithm, a robust method for efficiently fitting GMMs to data, presented by Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin [14]. Since then, various researchers have made constant efforts to apply and develop the method in the field of image classification [15-17].

2.2 The current state of deep convolutional learning networks

Deep learning comprises several branches, but the most effective for image recognition tasks are Convolutional Neural Networks (CNNs). The first notion of CNN was introduced in the late 1980s by Yann LeCun and his colleagues [21], but the turning point came in 2011 when AlexNet was presented [22], a model that marked the birth of deep neural networks. Apart from its eight layers, AlexNet utilized techniques that seem commonplace today, such as the Rectified Linear Unit (ReLU) activation function and dropout regularization, techniques with a significant impact on the network's accuracy. After this, an infinity of possibilities arose regarding the architecture of a CNN, but only some stood out through innovation. In 2015, Olaf Ronneberger, Philipp Fischer, and Thomas Brox introduced the U-Net architecture [25], an innovative and symmetric convolutional neural network oriented towards pixel classification. The initial aim of the study was biomedical image segmentation, but over time, U-Net has been adapted for use in a multitude of semantic segmentation tasks, such as handwriting recognition, medical observations, industrial automation, or satellite image segmentation, a domain under which deforestation monitoring falls under. Another reason for choosing U-Net over other architectures is its ability to function with fewer images and weak labeling.

Chapter 3

Database

Compared to other fields where images can be captured even with a mobile phone or a professional camera, collecting aerial images involves much higher resources and costs. Equipment such as photogrammetric systems or processing stations are expensive, and conducting and planning flights are complex activities. For this reason, most studies and research use open-source databases provided by government institutions, universities, or online communities. Even though they are free, most aerial or satellite image databases have relatively low radiometric resolution and are collected and possibly labeled for specific predetermined purposes, making their adaptation to other objectives challenging. Despite this, this doctoral thesis uses proprietary aerial images provided by the Defense Geospatial Information Agency for research purposes.

3.1 Description of the photogrammetric system

The aerial images were collected using a Leica Geosystems ADS80 (Airborne Digital Sensor 80) pushbroom digital photogrammetric system belonging to the Defense Geospatial Information Agency. The system was installed on a small Antonov An-30 aircraft, a model specially designed for aerial photography purposes. Besides the auxiliary elements on board the aircraft, the photogrammetric system consists of the recording sensor, the control unit, the storage memory for the recordings, and the software applications used to obtain the final product. Considering all these aspects, the aerial images produced had the following resolutions and characteristics:

- A spatial resolution of 50 cm, representing the size of one pixel on the ground;
- A spectral resolution of four bands: R (Red), G (Green), B (Blue), and NIR (Near-Infrared), indicating the wavelength range within which the image was recorded;
- An 8-bit radiometric resolution, corresponding to 256 possible grayscale values for recording the sensor's radiometric response.

3.2 Acquisition and processing of raw data

The information collected by the photogrammetric system consists of raw data, which is unusable for machine learning algorithms. This data underwent a complex processing procedure to obtain usable multispectral aerial images. The entire workflow is illustrated in Figure 3.3. After designing and executing the photogrammetric flight, the raw data was downloaded and visually inspected to identify clouds or birds that might compromise the clarity of certain areas. Following

a preliminary processing stage, the subsequent steps included aerotriangulation, orthorectification, and mosaicking.

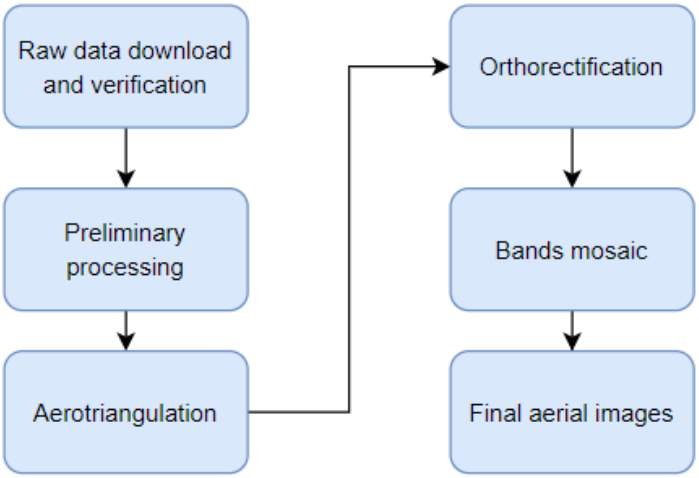


Figure 3.3 *Steps of processing raw data to obtain aerial images*

3.3 Obtaining the final database

Considering the institution's interests, the processed data has been exported to obtain two types of products: RGB (Red Green Blue) aerial images and CIR (Color InfraRed) images. Additionally, the flight area was divided into 10 × 10 km2 zones and saved as TIFF (Tagged Image File Format) files. To support the research for this doctoral thesis, AIGA provided eight such zones, four near the city of Târgu-Mureș and four near the commune of Roșia Montană, areas rich in forests but also mixed with other land cover classes such as human settlements, crops, pastures, or roads. In Figure 3.5, an example of a 100 km2 aerial image in both RGB and CIR formats can be visualized.

(a) (b)

Figure 3.5 *Samples of aerial images; (a) RGB image; (b) CIR image*

To eliminate this redundancy and retain only the relevant information, the near-infrared band was extracted from the CIR images and concatenated with the RGB images, resulting in RGBN images. Finally, the three datasets RGBN, RGB, and NIR were sliced into three dimensions of 4000 × 4000, 1000 × 1000, and 200 × 200 to introduce diversity before applying the learning algorithms. This generated a generous dataset, eliminating the need for augmentation used by many other studies in the field.

Chapter 4

Study of unsupervised algorithms for identifying terrestrial details

The main idea for achieving the assumed objectives was as follows: to develop a flexible solution capable of ingesting aerial images collected under various conditions and to circumvent the need for their labeling, a clustering technique can be used, the result of which can serve to train a supervised network or be improved and used autonomously. This approach shortens prediction time by eliminating the photointerpretation step, and the developed algorithm will not overfit a particular dataset. To achieve the most efficient labeling, four clustering algorithms were selected for application to aerial images: K-Means, AGNES (Agglomerative Nesting), GMM, and Mean Shift. At the end of the chapter, it was concluded which of these four techniques yielded the best results and was chosen as the basis for further developments.

To perform a manual analysis of the content of the processed images, a team of GIS experts specialized in photo interpretation was employed, following the workflow illustrated in Figure 4.1. They established five classes representing the percentage of forest area possibly present in an image, as follows: completely forest-covered image (100%), large coverage (75%), medium coverage (50%), small coverage (25%), and image without forest (0%). Then, each image used as input data was categorized into one of these classes. Subsequently, for each segmented image, the amount of forest identified by the clustering algorithms was calculated and rounded to the nearest class among the five. Since the total number of pixels is known, this could be achieved through a simple proportion, after counting all forest-containing pixels. Finally, accuracy was calculated by comparing the two labels: the input image label established by experts and the label calculated through counting.

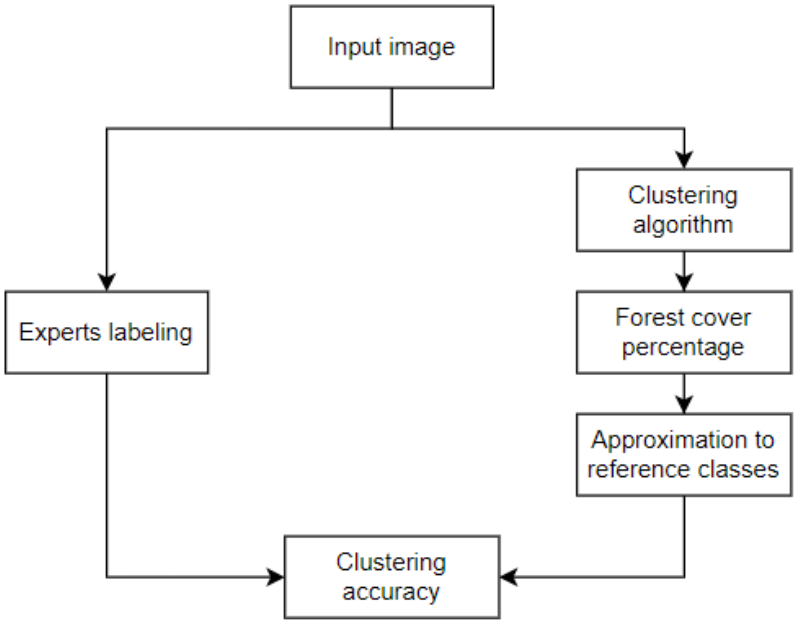
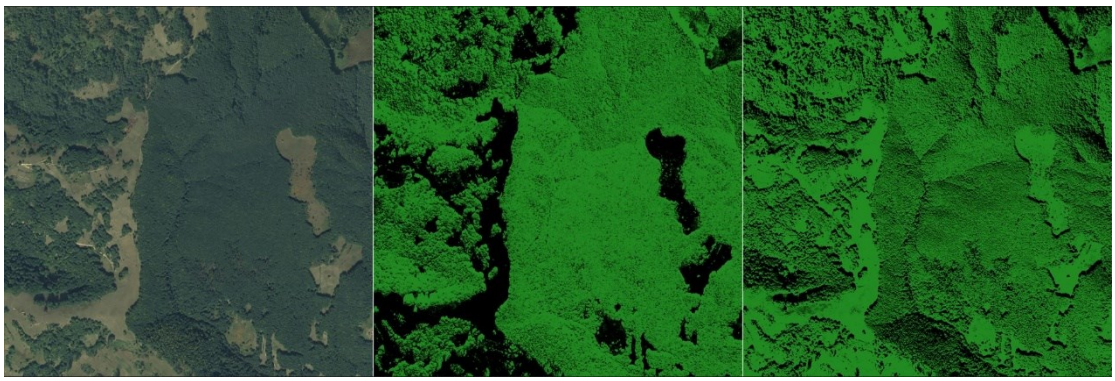


Figure 4.1 *Diagram for assessing the accuracy of clustering algorithms*

4.1 K-Means

K-Means clustering was applied to both RGB and RGBN images. Surprisingly, the RGB images performed very well without any preprocessing, while the addition of the extra NIR band yielded weaker results. This situation is highlighted in Figure 4.5. The assumption for this phenomenon is that the near-infrared band contains high values for all healthy vegetation on the ground, not just forests, and the clustering algorithm could not distinguish this aspect during distance calculation, considering, for example, pasture as forest-covered areas.



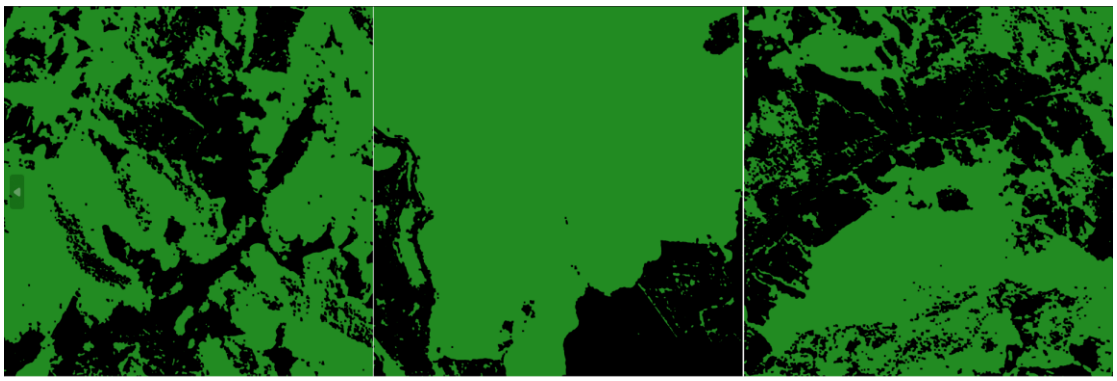
(a) (b) (c)

Figure 4.5 *Results of K-Means clustering; (a) ground truth; (b) segmented RGB image; (c) segmented RGBN image*

Two hundred aerial RGB images sized 4000 × 4000 were utilized. These images were labeled by GIS experts into the five established classes, and following clustering, an accuracy of 85.56% was calculated. This score was deemed satisfactory

as a starting point and a good basis for further implementations and improvements (Figure 4.7).

(a)



(b)

Figure 4.7 *Examples of K-Means clustering; (a) input images; (b) segmented images*

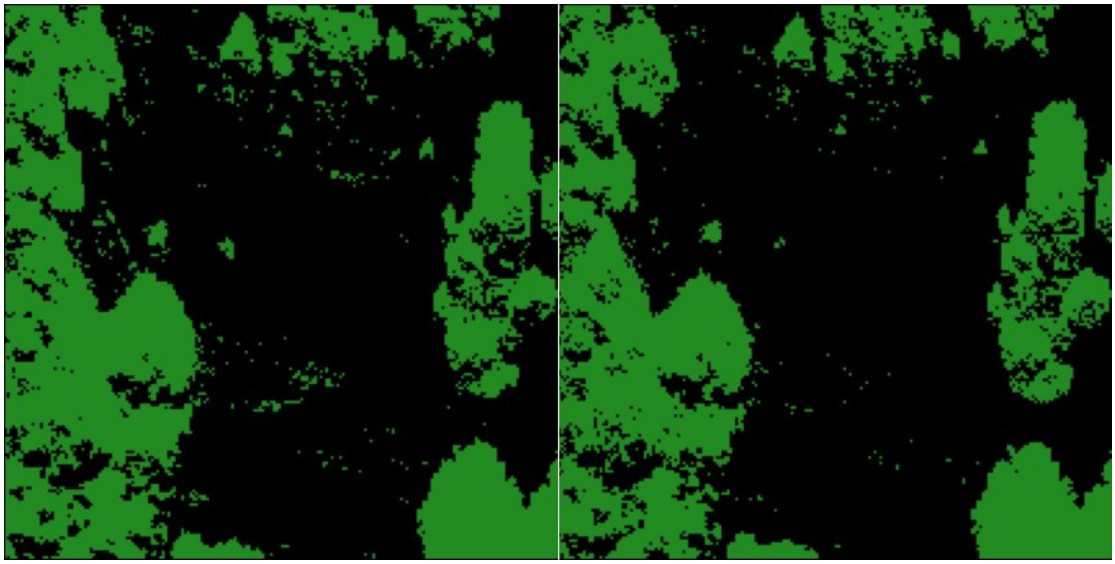
4.2 Agglomerative clustering

Due to computational resource constraints, the maximum size at which the AGNES hierarchical algorithm could be applied to the NIR, RGB, and RGBN datasets was 200 × 200. Metrics used included L1, L2, and cosine distances, along with linkage criteria of single, complete, average, and Ward's method. The only scenarios that yielded satisfactory results were the complete linkage criterion using L1 and L2 metrics and Ward's method. Complete linkage performed better using all four spectral bands, while Ward's method showed better results using the RGB dataset.

Two hundred aerial images of size 200 × 200 were employed, and the final accuracies for each scenario were as follows:

- Complete linkage, L1 norm, RGBN: 58.33%;
- Complete linkage, L2 norm, RGBN: 73.33%;
- Ward's method, L2 norm, RGB: 87.78%.

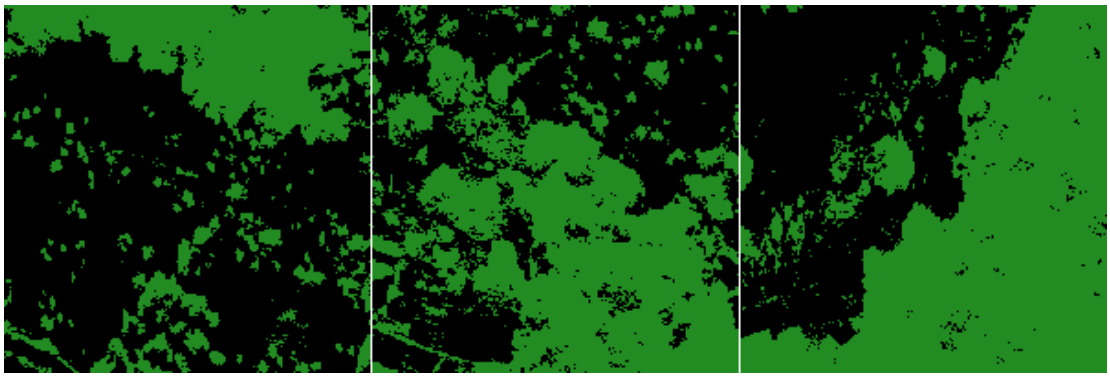
Comparing Ward's method results for the RGB dataset (examples in Figure 4.12) with the complete linkage results using the L2 norm for the RGBN dataset, a clear improvement was observed. The image exhibited less noise, and the forest edges were much better defined (Figure 4.11). Appendix 2 provides further magnified results of Ward's method applied to RGB images.



(a) (b)

Figure 4.11 *Comparative results of the L2 norm; (a) complete linkage criterion; (b) Ward's method*

(a)



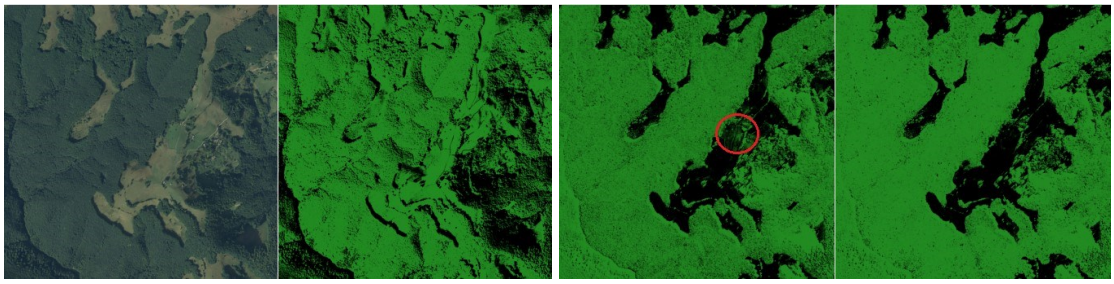
(b)

Figure 4.12 *Samples of AGNES clustering; (a) input images; (b) segmented images*

4.3 Gaussian mixture model

In previous studies where K-Means and AGNES algorithms were implemented, the results of the RGB dataset proved to be better than the results of the RGBN dataset. However, with GMM, the outcomes were different. Applying GMM

to the NIR band yielded predictable results, as it was not feasible to construct a mixture with such limited information. The results of the other two datasets, at first glance, appeared similar, but the RGBN images exhibited slightly better performance.



(a) (b) (c) (d)

Figure 4.15 *GMM results for images with different depths; (a) ground truth; (b) NIR; (c) RGB; (d) RGBN*

Another important factor of any machine learning algorithm is the size of the input images or the analysis window based on which calculations are made. Testing began with an initial image size of 20,000 × 20,000, and later the input size was reduced to smaller areas, namely 4000 × 4000, 1000 × 1000, and 200 × 200. Along with analyzing the results, computation time was also measured. It is presented in Table 4.1 using seconds as the unit of measurement. The number of inputs represents how many lower-order images compose a higher-order image. For the same area and the same input data, comparisons must be made diagonally. Taking all these factors into consideration, the best two input sizes were chosen: 4000 × 4000 and 1000 × 1000.

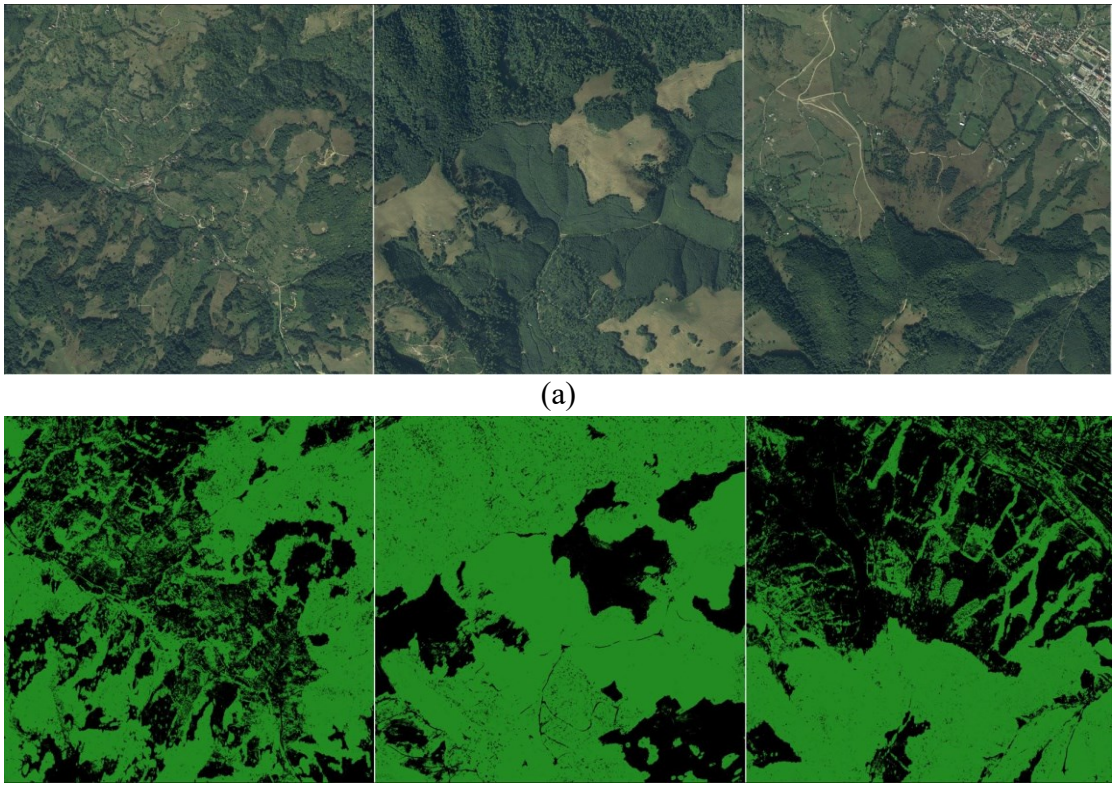
Table 4.1 *Computation time of GMM for different input dimensions*

Dimension	20000	4000	1000	200
No. of inputs	-	25	16	25
Average time (s)	17745.2750	151.7057	11.1756	0.6246
Total time (s)	-	3792.6430	178.8102	15.6150

In the end, the 200 input RGBN images and the clustering results were presented to the GIS expert team for analysis. Following the labeling of the images into the five classes, the final accuracy of GMM for the four considered scenarios is presented in Table 4.2. The best results were obtained using input data of size 4000 × 4000. In this case, both initialization methods performed very well, and the difference between their results consisted of a small number of pixels classified differently, which did not have a real impact on the expert evaluation and final accuracy.

Table 4.2 *The accuracy of the GMM algorithm*

Dimensions / Initialization	**1000 × 1000**	**4000 × 4000**
Random	86.11	92.22
K-Means	88.89	92.22



(a)

(b)

Figure 4.18 *Examples of GMM clustering; (a) input images 4000 × 4000; (b) segmented images*

4.4 Mean Shift

If, in the case of the other three algorithms, each image was segmented independently, with a training stage where the centers of each cluster were identified, and a prediction stage where the entire image was segmented, for Mean Shift, it was decided to use a single image from which the centers of the two clusters were extracted, and then the other images were segmented using these two centers. For testing and accuracy calculation, 400 images of size 1000 × 1000 from the RGBN dataset were used.

In addition to the well-documented theoretical basis of the Mean Shift method [67], the implemented algorithm was modified and augmented with certain elements. The implementation of the entire method is illustrated in Figure 4.20. First, the training image is introduced into a module that estimates the most important parameter of the algorithm, namely the bandwidth. Secondly, the training image undergoes an iterative process that provides a series of centroids. After removing duplicate centroids, only the centroids of the two classes of interest remain. Then, these centroids are used to predict forest and non-forest areas for the entire dataset.

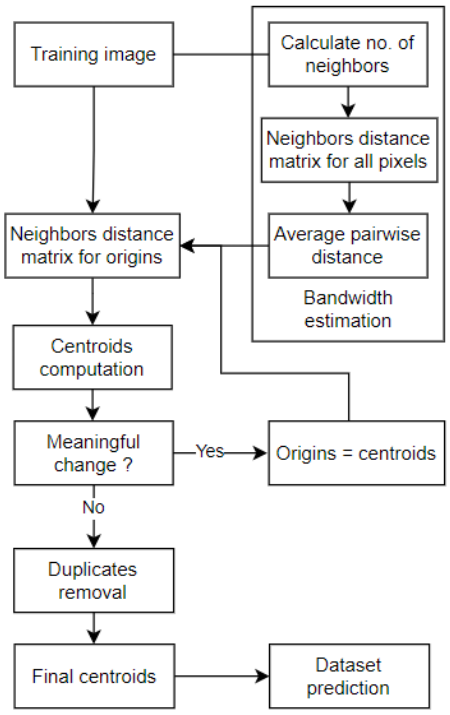


Figure 4.20 *Mean Shift diagram*

The algorithm requires two adjustable parameters to ensure convergence: the scaling factor Q and the maximum number of iterations T. After running several scenarios, it was determined that Q = 0.2 and a bandwidth B = 25.38 were optimal for this dataset. Three threshold values for the scaling factor were considered: 20, 30, and 50. Following evaluation by GIS experts, the Mean Shift algorithm achieved good results, close to 90%:

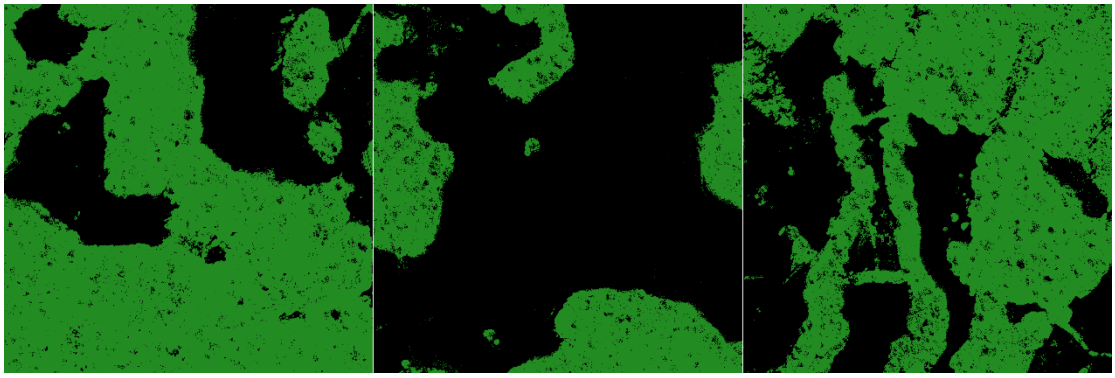
Mean Shift 20: 89.75%;

Mean Shift 30: 90.75%;

Mean Shift 50: 89.75%.

Figure 4.24 presents some segmented images for the best-identified scenario, namely Mean Shift clustering with a scaling factor of Q = 0.20 and a threshold for the maximum number of iterations T = 30.

(a)



(b)

Figure 4.24 *Samples of Mean Shift clustering; (a) test images; (b) segmented images*

4.5 Conclusions

Due to the distinct characteristics of the methods, they could not be uniformly applied for the same areas of interest or input data size. They were studied independently, so for a comprehensive evaluation by GIS experts, a more detailed comparison was necessary to determine which technique performed better. To achieve this, computation time and memory allocation of the system were measured for each computational step, including data preprocessing, training, and prediction.

To fairly compare the methods, measurements were scaled to the largest input size, that of the GMM clustering of 4000 × 4000 × 4. Specifically, using the input dimensions and the number of bands of the other three methods, a scaling factor was calculated to multiply all measured values (Table 4.9). Overall, the least performing method was AGNES, with values significantly higher than the others. Despite achieving over 90% accuracy, Mean Shift required approximately 2037.72 seconds and 718.4 Mb of memory to classify an image of size 4000 × 4000 × 4. The choice of the most performant technique thus boiled down to a comparison between K-Means and GMM. Although GMM required more than twice the time compared to K-Means, it performed much better in the other two categories, obtaining the lowest memory allocation of only 492.6 Mb and the highest accuracy of 92.22%, nearly 7 percentage points higher than K-Means. Thus, it was established that the clustering method underlying the subsequent study would be GMM.

Table 4.9 *Scaled performance of the clustering algorithms*

Method	**Total time (s)**	**Total memory (Mb)**	**Accuracy (%)**
K-Means	26.91	534.66	85.56
AGNES	34549.84	2447.2	87.78
GMM	61.89	492.6	92.22
Mean shift	2037.72	718.4	90.75

Chapter 5

CEM – Color extraction module

Due to the absence of labeled data, unsupervised classification cannot automatically associate the clustering results with the real classes of membership, thus making their representation accurate. This aspect of clustering techniques has posed a challenge for manipulating and interpreting results during the study, as it has hindered the ability to present intuitive information for a lay user and slowed down the evaluation process. The development process of the CEM had to be automated in such a way that there is a link between clustering methods and input images. To serve purposes other than deforestation monitoring, it was established that the CEM should be independent of the implemented methods or their parameters and be easy to apply outside of the current study. The diagram of the CEM algorithm is presented in Figure 5.2, generalized for an infinite number of classes. In real applications, the number of classes can vary from 2, if specific elements need to be identified (lake, forest, human settlement, etc.), to many more for advanced land cover segmentations.

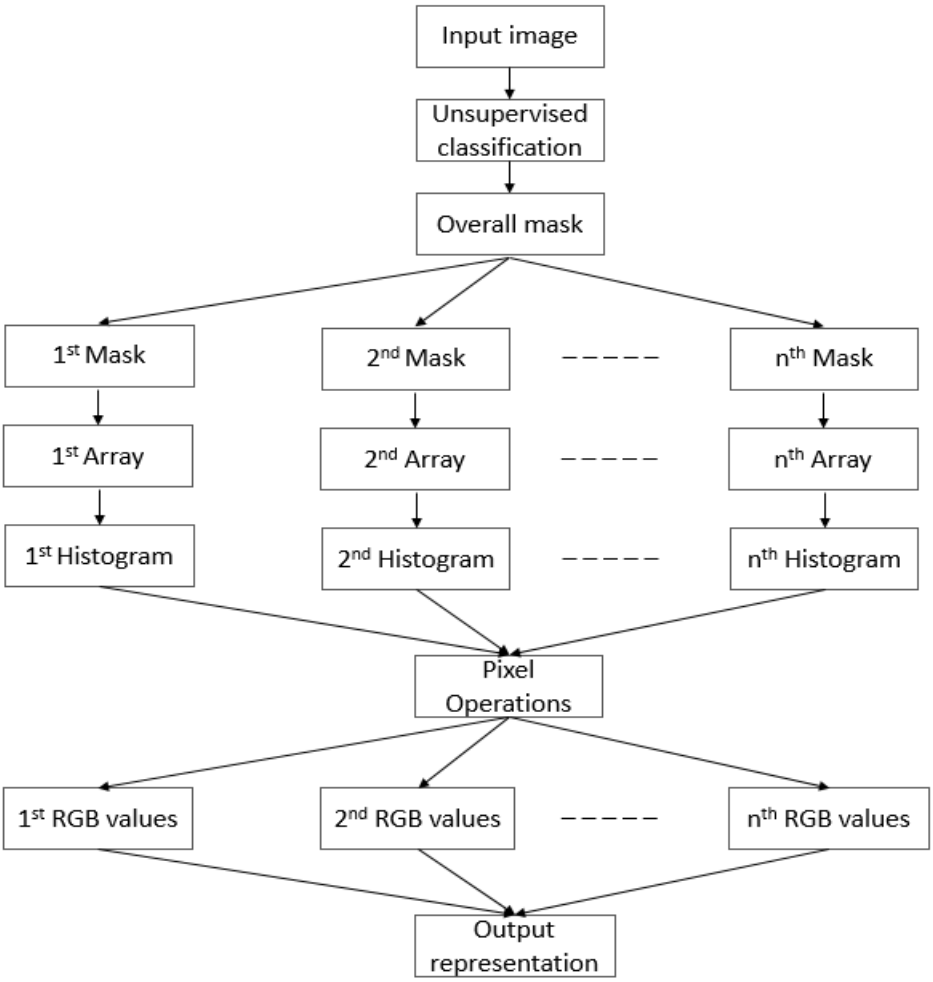


Figure 5.2 *CEM Diagram*

After the training stage, each unsupervised algorithm calculates its own set of variables describing the classes. However, the results are quantified using a common attribute, namely the label assigned to each pixel. The matrix formed by the total labels constitutes the general mask of the image. This mask is then divided into n binary masks, one for each class identified after segmentation. In the next step, each of these masks is overlaid on the input image, forming n vectors of values describing each class. Although these n vectors are not traditional images, they are composed of pixel values, so their histograms are calculated. In the subsequent pixel operations stage, three averaging functions and two counting functions were implemented and compared to determine which best suited the purpose of the CEM.

Ultimately, the three RGB values are used to represent each of the identified classes. The choice of the best function was established through visual and numerical comparison. Figure 5.6 illustrates the final representation of the two classes. The two classes are similarly represented by all five functions, with the general mode function having slightly better contrast than the others. Compared to the three means, which have values closer between the two classes, the mode functions extracted more eccentric values. In the example below, the non-forest class consists mostly of grassland and was illustrated with a lighter green than the forest, indicating that the road and the few human elements identified in this class did not have a major effect on the final representation.

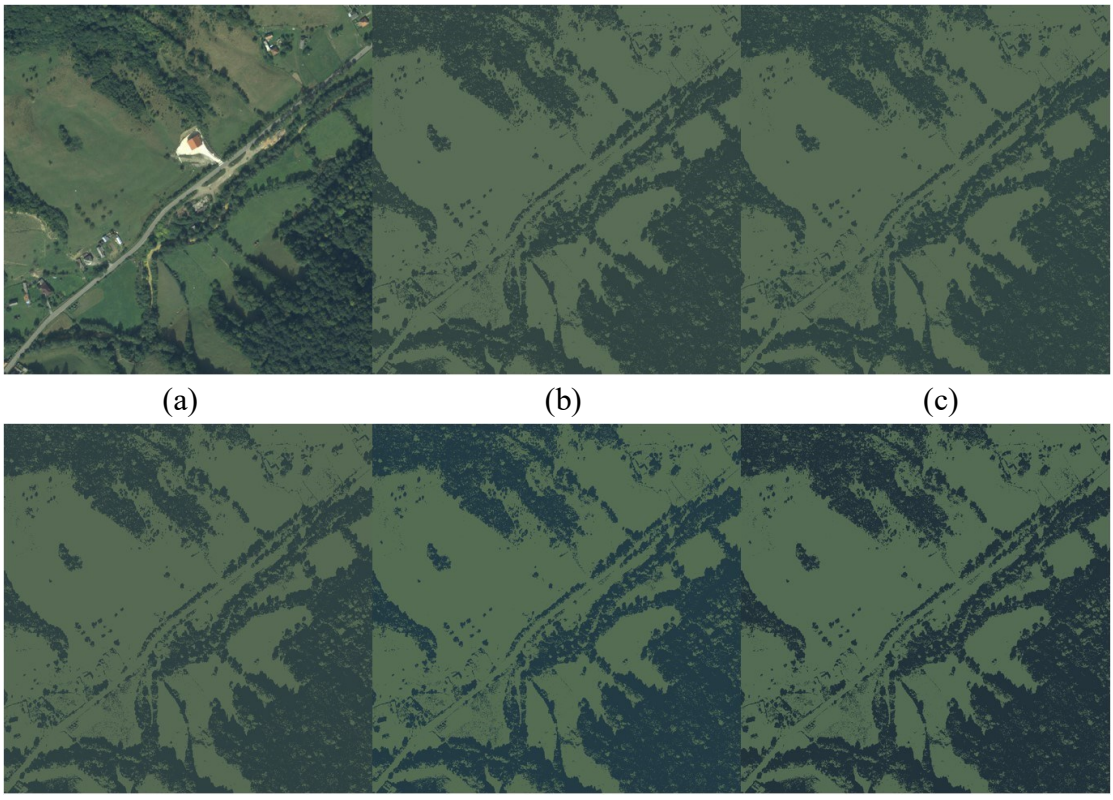
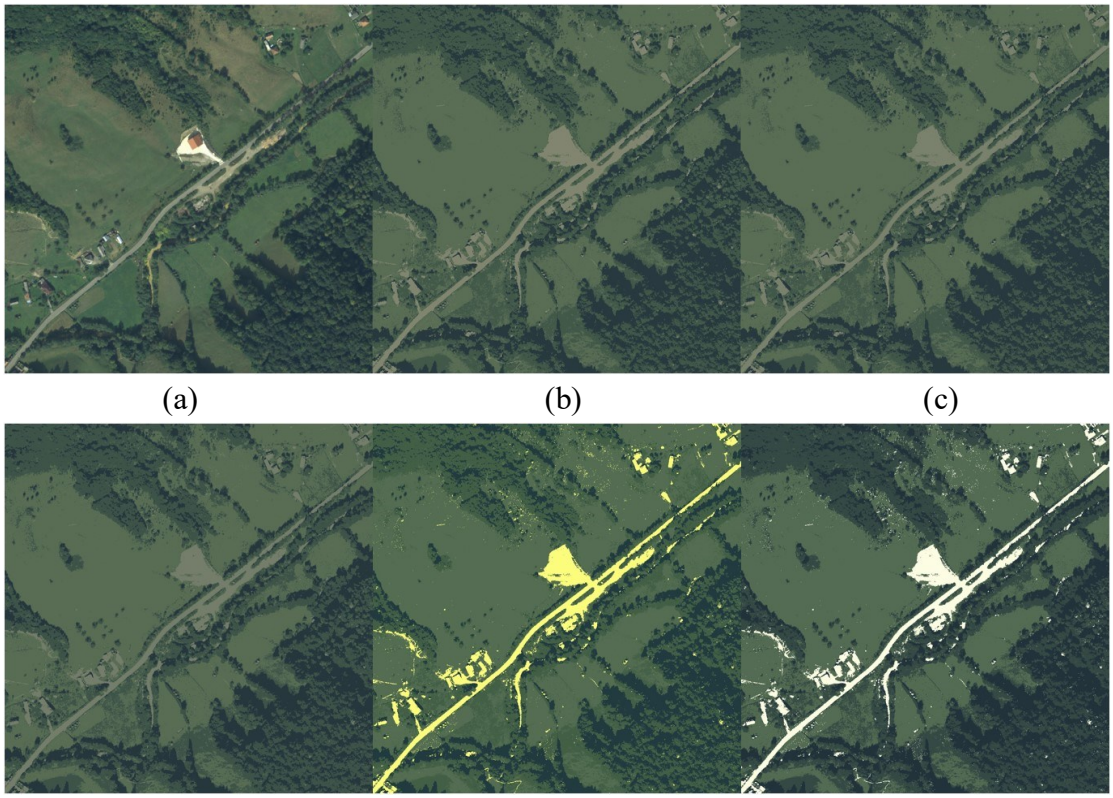


Figure 5.6 *Representation of two-classes clustering; (a) Ground truth; (b) AM; (c) GM; (d) HM; (e) Local mode function; (f) Global mode function.*

On the other hand, the final representation for 4 classes is shown in Figure 5.7, and Table 5.2 presents the calculated values for each function. As can be observed, changing the number of classes to be identified for a clustering algorithm significantly altered their display mode for counting functions. The three means again had similar results and representations. For all five functions, the three classes representing grasslands, shadows, and forests were illustrated in a very intuitive manner, closely resembling their actual colors. The main difference between methods was constituted by the RGB values of the infrastructure class, which encompasses human elements in the image. This was illustrated by the local mode function in yellow, as the calculated blue value was much lower than red and green. The general mode function depicted it in off-white (253,253,227) due to the area around the central house, which constituted the majority of pixels. On the other hand, the mean functions leveled this area with the rest of the road and rooftops, obtaining different shades of gray. These are obtained when the RGB values are similar, as highlighted in Table 5.2.



(a) (b) (c)

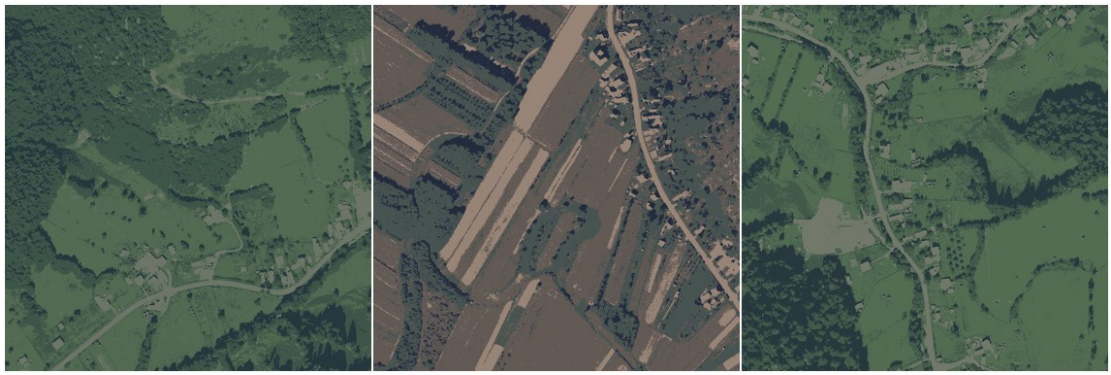
(d) (e) (f)

Figure 5.7 *Representation of four-classes clustering; (a) Ground truth; (b) AM; (c) GM; (d) HM; (e) Local mode function; (f) Global mode function.*

Comparing the differences in representation between two and four clusters, it was concluded that the largest deviations from reality will occur in classes with very high diversity, such as the infrastructure class, which can contain houses with different roofs, roads made of different materials, or various civil constructions. Taking all of the above into account, it was established that the safest and least unpredictable function for the pixel operations stage is one of the averaging functions, specifically the general mean (GM). The local mean (LM) represents classes too

darkly, while the adaptive mean (AM) contains even shades of yellow in some instances. CEM was implemented over the clustering methods, and its development significantly aided in accelerating the research and the overall study, especially in the visual interpretation and evaluation of the clustering results. Various representations for 4 classes are illustrated in Figure 5.9.

(a)



(b)

Figure 5.9 *Examples of final representations using CEM; (a) Ground truth; (b) CEM representation of clustering into four classes.*

Chapter 6

GMM improvement through compression and signal processing techniques

After implementing and comparing the four clustering techniques, at the end of Chapter 4, it was concluded that the best basis for developing a methodology to achieve the assumed objectives is Gaussian Mixture Model (GMM) clustering. However, the major problem identified in applying GMM was the long computation

time and large memory allocations, regardless of the native optimizations of the algorithm. To address this issue, methods of image compression and signal processing were studied. Thus, Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT), and Discrete Wavelet Transform (DWT) were applied to the input images before GMM clustering.

Traditionally, discrete transformations are applied to smaller regions of the image to improve the feature extraction step. The methodology developed in this study proposes a different approach, illustrated in Figure 6.2, by using an extraction window for the entire image. First, the input image undergoes a decomposition stage into bands. Then, the three discrete transformations are calculated for each band in a two-dimensional space. To extract only the significant coefficients, an extraction window is applied over the transformed image. The window position depends on the type of transformation. For DCT, the most important coefficients are concentrated in the bottom-left corner of the transform, while DFT, after shifting the zero frequencies, concentrates them in the middle of the image. Several scenarios were created by modifying the size of the extraction window to determine the best value for the study objectives and to analyze the trade-offs in clustering quality and costs. Then, the extracted coefficients for each band are converted back using inverse transformations. It's worth noting that these extracted coefficients are not padded with zero values to maintain the original length and width of the image, meaning dimensionality reduction is achieved in this step. For DWT, these two extraction and inverse transformation steps do not exist, as this transform naturally extracts approximation coefficients, reducing the size of the images by half. Thus, there was no need to implement an extraction window and an inverse transformation.

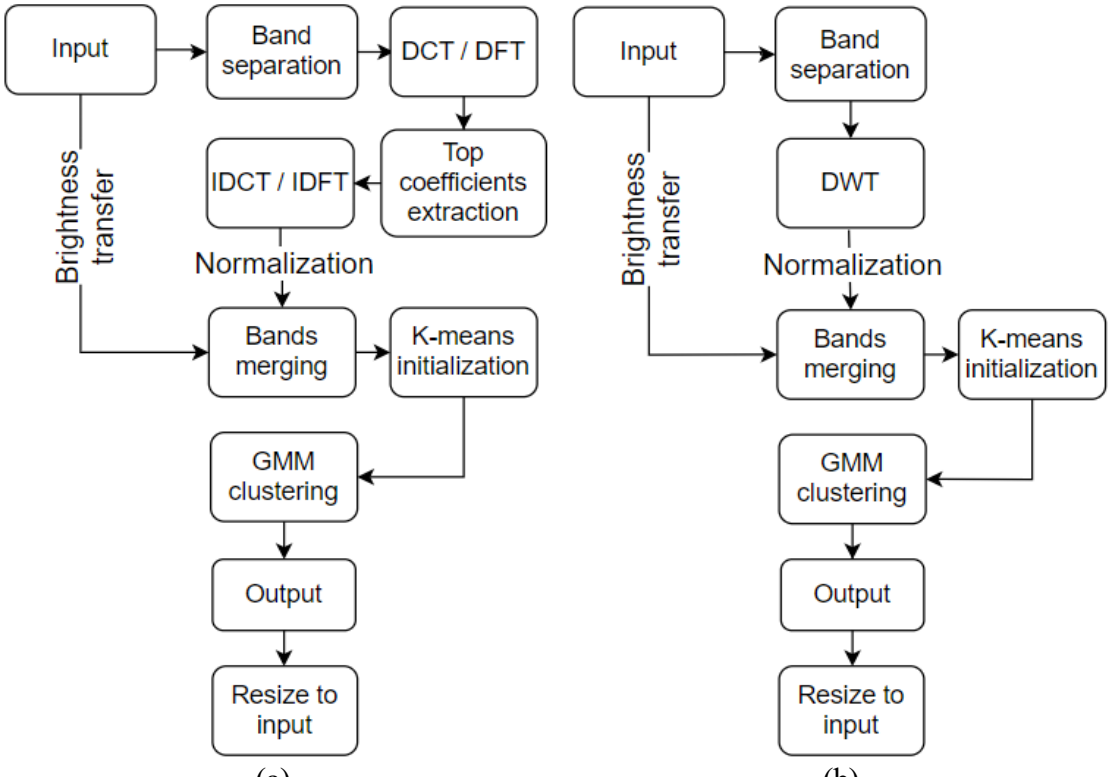


Figure 6.2 *Compression algorithm diagram: (a) DCT / DFT; (b) DWT*

After applying normalization and brightness transfer to the results of discrete transforms, the bands are concatenated and undergo a fast K-Means clustering stage. This serves as initialization for the GMM algorithm. Its speed comes from establishing the initial centroids probabilistically, not randomly, by calculating the distance between points. Finally, the results of GMM clustering are resized to the size of the original input using nearest neighbor interpolation.

6.2 Evaluation of the results of the proposed methodology

The proposed methodology was tested within a clustering of five classes using a generous dataset of 1600 images with dimensions of 1000 × 1000 × 4. To evaluate the results, we analyzed the proposed method from three perspectives: computational time, allocated memory, and performance. Unfortunately, for unsupervised learning algorithms, accuracy cannot be calculated in the true sense of the word because the data is not labeled. For this reason, performance was evaluated through the Davies-Bouldin index (DBI). Computation time, allocated memory, and DBI scores were measured for each extraction window, as well as for the conventional, unmodified GMM algorithm. All values were transformed into percentages of the reference GMM algorithm.

Thus, the percentage of time and memory should have been as low as possible, while the DBI percentages should have been as close to 100% as possible. Although all three 500 scenarios had small decreases in DBI score and faster calculation times, the results of DFT 500 and DWT 500 were neglected due to memory increase. DCT 500 had a computation time of 32.82% while maintaining clustering quality and total allocated memory. For the second group of 250, the DBI score decreased by approximately 10%, but the total time and memory showed substantial improvements. This time, DCT 250 and DFT 250 had similar percentages and outperformed DWT 250. While DTW 250 had a better DBI score, the difference of only 2% was not significant enough to outweigh the much larger memory allocation of 78.17%. Unfortunately, despite reduced computation times, the 125 scenarios experienced an approximate 20% decrease in DBI scores. Overall, DCT performed better than the other two transformations, mainly because it provides information in the form of a single set of coefficients. DFT obtains both real and imaginary coefficients, while DWT provides an approximation image and three detail images.

Table 6.3 *Percentage analysis of GMM scenarios performance*

Scenario	DBI (%)	Total time (%)	Total memory (%)
DCT 500	95.35	32.82	93.53
DFT 500	94.31	31.42	115.63
DWT 500	95.89	32.02	109.97
DCT 250	89.76	8.94	53.37
DFT 250	89.00	9.26	56.87
DWT 250	91.78	8.89	78.17

DCT 125	80.99	2.78	41.78
DFT 125	79.02	3.22	42.59
DWT 125	87.66	3.13	66.04

Comparing DCT 250 and DFT 250, we can observe that DCT 250 had slightly better performance than DFT 250 in all three measurements. Taking into account all the information presented above, we can assume that the best scenarios of the proposed method are DCT 500 and DCT 250. Depending on the implementation conditions, both can be suitable and worth considering. Figure 6.8 depicts the input image and the results of the two scenarios.

(a) (b) (c)

Figure 6.8 *Results of the best scenarios: (a) Ground truth; (b) DCT 500; (c) DCT 250*

6.3 Conclusions

This study aimed to implement a rapid and cost-effective unsupervised algorithm to keep pace with deforestation phenomena and provide real-time analysis and alerts to relevant authorities by improving a previously analyzed clustering method applied to the same dataset. The study focused on reducing the computation time of the GMM clustering method, previously used for forest segmentation, by proposing an algorithm that leveraged the advantages of discrete transformations conventionally used in image compression and signal processing.

The proposed algorithm was tested using a dataset of 1600 aerial images of 1000 × 1000 × 4 dimensions and spatial resolution of 50 cm each, by measuring the computation time, DBI score, and allocated memory for each considered scenario. All results were compared with the previous GMM algorithm, which served as a reference baseline for comparisons, with its performance measured under the same conditions. The scenarios exhibited much-improved computation time, but only DCT 500, DCT 250, and DFT 250 maintained a DBI score close to the reference GMM algorithm while also improving memory allocation. On average, the GMM algorithm had a total computation time of 19.9158 s and a memory allocation of 37.1 Mb. DCT 500 managed to reduce these costs to 6.5360 s and 34.7 Mb, with a minimal decrease in the DBI score.

In conclusion, this methodology achieved the study's objectives and can be considered a viable solution for classifications with limited resources. It can be strongly affirmed that the described algorithm solves the problem of computation time and even the memory allocated by the computing unit.

Chapter 7

Supervised learning using the U-Net architecture

7.1 Automatic data labelling

This study proposes an approach using GMM clustering for the semantic labeling of images before training U-Net architectures for forest classification. To establish the dataset, the conclusions of Chapter 4 were applied, namely the use of 100 multispectral aerial images of 4000 × 4000 × 4 from the Târgu-Mureș area. GMM segmentation is augmented by several techniques to improve labeling, such as computing informational criteria, merging clusters, and filtering resulting images. After applying GMM, Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) scores are calculated to validate the optimal number of clusters. Then, the resulting clusters are merged to form only the two classes of the study: forest and non-forest. After filtering the results, the DBI score is used to select the best-labeled images, thus constructing the training, validation, and testing datasets for U-Net. Finally, the labeled dataset is divided into smaller batches of 128 × 128 and fed into U-Net architectures composed of 19 convolutional layers and 4 skip connections. Figure 7.7 illustrates samples of the labeled dataset.

(a)

(b)

***Figure** 7.7 Examples of automatic labeling: (a) Training images; (b) Labels*

7.2 Description of the U-Net architecture

U-Net utilizes an encoder-decoder architecture, composed of two main pathways. The encoding pathway systematically reduces the dimensions of the input image, thereby extracting higher-level features. On the other hand, the decoding pathway restores encoded features through upsampling, leading to an enhanced segmentation mask with high spatial resolution. This unique architecture enables the network to grasp both the broader context and local details during the learning process. The original U-Net architecture consists of an input layer, four encoding blocks, a bridge, four decoding blocks, four skip connections, and an output layer (Figure 7.8). U-Net uses the ReLU (Rectified Linear Unit) activation function for all layers in the model, except the final layer, which requires a sigmoid function. Additionally, a dropout layer is employed at the end of the encoding pathway to prevent overfitting of the model.

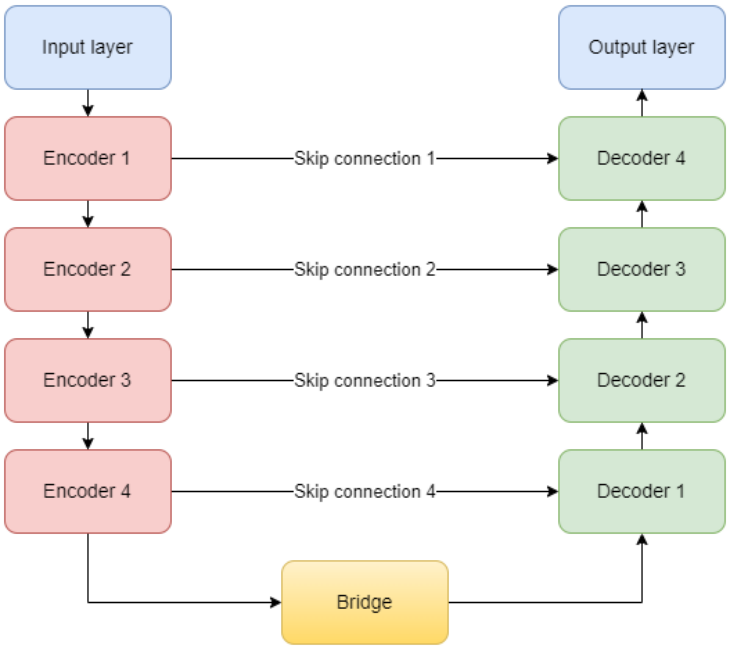


***Figure** 7.8 The original U-Net architecture*

7.3 *Optimization of the non-structural hyperparameters*

The multitude of hyperparameters of a CNN can be divided into two groups. Structural hyperparameters are those presented in the previous section, namely: the number of convolutional layers, the number of kernels, the kernel size for each convolution, activation functions, and pooling dimensions. To adjust the other non-structural hyperparameters, computational power must be taken into consideration. To address this issue, the batch size for training, which determines the number of samples processed in one forward and backward pass for each iteration, was incrementally indexed until the system returned an Out of Memory (OOM) error. This way, the maximum supported value by the computing unit could be determined. In this analysis, the model was set to compute only one epoch, and the computation time for that epoch was monitored. The results are presented in Table 7.1. Thus, the maximum batch size accepted by the GPU was 16, which also had the best training time according to the table.

Table 7.1 *Computation time for different training batch sizes*

Batch size	2	4	8	16	32
Computation time (s)	1502	1112	805	520	OOM
Validation accuracy	0.9897	0.9912	0.9914	0.9903	OOM

Having such a high initial accuracy, it was concluded that the rest of the adjustments should be made through fine-tuning over 10 epochs. Setting the batch size and optimizer directed us to the next hyperparameter to be adjusted, namely the learning rate. The best validation accuracy of 0.9951 was achieved for a learning rate of 10^{-4}. Additionally, for this value, the validation accuracy was kept closer to the training accuracy throughout the 10 epochs.

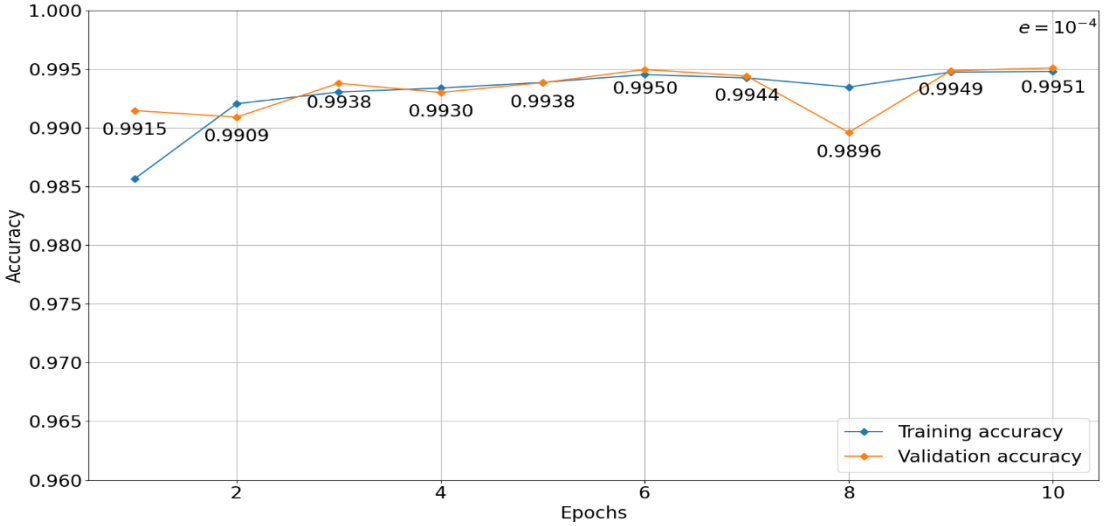


Figure 7.10 *Micro-training accuracy of U-Net*

7.4 U-Net model cost reduction

Unfortunately, the original architecture, as presented earlier in Section 7.2 and as initially proposed in [25], was far too complex for the objectives of this study. With all non-structural hyperparameters adjusted, it was decided to simplify the U-Net architecture by systematically reducing the number of filters in each convolutional layer by half. Thus, multiple scenarios were created while maintaining the symmetry and indexing rules of the original architecture.

To analyze the performance of the simplified models in detail, the allocated memory, accuracy, and computation time were measured for a single random image. For a better understanding of the improvements in the proposed scenarios, the measurements were graphically represented in Figure 7.12. The corresponding values for the graph are found in Table 7.5 and represent percentages of the original U-Net scenario. Thus, a successful scenario will be described by accuracy as close to 100% as possible, along with reduced time and memory allocation. The graph has two vertical axes, accuracy and time projected on the left axis, while memory is projected on the right. It can be strongly asserted that all scenarios yielded very good results. Accuracy maintains high percentages while time and memory consistently decrease until the point where S-UNet2 calculates twice as fast and utilizes only 3.38% of the initial memory. Analyzing the two exposures, it can be observed that S-UNet8 has the best balance between accuracy and reduction of time and memory. It managed to maintain 99.91% of the accuracy of the original U-Net model using only 4.38% of its memory and having a computation time almost twice as fast.

Table 7.5 *Percentage analysis of performance for simplified U-Net models*

Model	Accuracy (%)	Time (%)	Memory (%)
S-UNet32	99.98	74.71	27.25
S-UNet16	99.99	61.16	8.56
S-UNet8	99.91	56.56	4.38
S-UNet4	99.45	55.44	3.83
S-UNet2	99.45	51.71	3.38

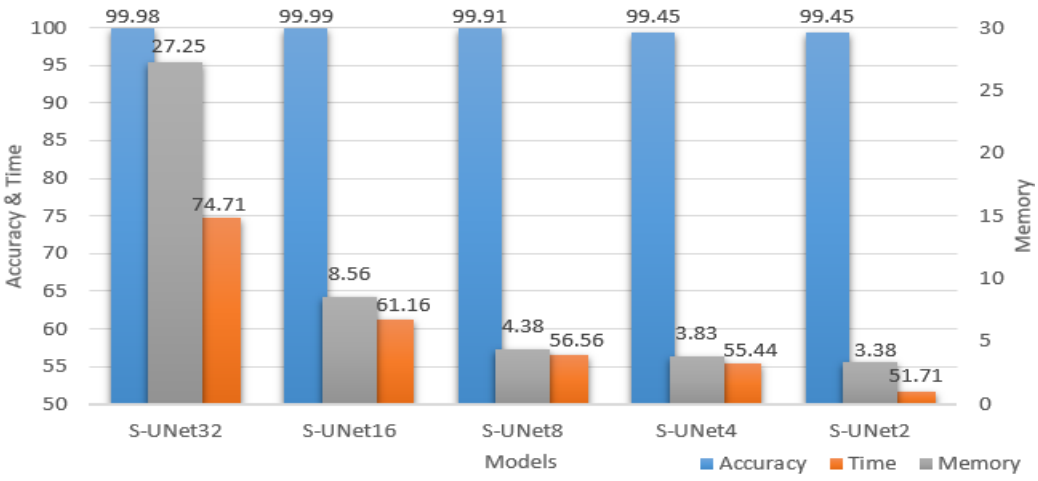


Figure 7.12 *Performance graph for simplified U-Net models*

Regarding the cost reduction of the proposed method, a much more complex analysis can be conducted. Table 7.7 describes the memory usage, total number of parameters, and total number of filters for LCU-Net [41], Half-Unet [43], and the simplified U-Net models. Even though LCU-Net has the same number of blocks and filter growth as the third scenario, from 16 to 256, due to the implementation of the Inception technique, the total number of filters and parameters is much higher. On the other hand, Half-Unet has 6 blocks with 64 filters each, as it disables 3 out of the 4 decoder blocks, resulting in only 212,576 parameters. Except for S-UNet32, the proposed method yielded better results in terms of cost reduction for all other models. S-Unet16 managed to use only 55.7 Mb compared to the needs of LCU-Net and Half-Unet, which are 103.5 Mb and 137.3 Mb, respectively.

Regarding the average time, this metric can be strongly influenced by hardware resources. Data for Half-Unet were not presented, but LCU-Net predicted results on average in 0.15 seconds using an Intel(R) Core(TM) i7-8700 processor with 3.20 GHz, 32 Gb RAM, and an Nvidia GeForce RTX 2080 graphics card with 8 Gb. In addition to the fact that the simplified U-Net models had much better prediction times, below 0.10 s, the computing unit used in our case also had a less powerful graphics card, which would lead to even faster results in the case of similar resources available with LCU-Net.

Table 7.7 *Comparison with related studies on costs and complexity of models*

Model	Memory (Mb)	Average prediction time (s)	Total no. of filters	Total no. of parameters
LCU-Net	103.5	0.15	5392	3469393
Half-UNet	137.3	-	768	212576
U-Net	650.7	0.142	6848	31032321
S-UNet32	177.3	0.106	3424	7760385
S-UNet16	55.7	0.087	1712	1941249
S-UNet8	28.5	0.081	856	485889
S-UNet4	24.9	0.079	428	121761
S-UNet2	22	0.074	214	30585

7.5 Conclusions

This chapter addressed a common problem in training neural networks, the lack of labeled data, and the disadvantages of obtaining them. To solve this problem, the GMM clustering method was used for automatic data labeling, and a workflow was implemented to obtain a consistent dataset for the supervised training of a convolutional neural network to identify forests in aerial images with very high accuracy. The consistency of the dataset was validated by training CNN models based on the U-Net architecture. Benefiting from the quality of the dataset, the accuracy achieved during training, validation, and testing was extremely high. Non-structural hyperparameters, such as learning rate, optimizer, or batch size, exhibited unusual

flexibility, demonstrating that the GMM clustering step further improved supervised training.

The original U-Net model achieved a validation accuracy of 0.9951 and a testing accuracy of 0.9969, just after fine-tuning for 10 epochs. In conclusion, it can be strongly asserted that the GMM method aids in data labeling and augmentation and reduces the computational resources required for supervised CNN training for rapid and inexpensive deforestation monitoring. Regarding hyperparameter optimization, the study showed that an S-UNet8 model with 856 filters achieves more than satisfactory accuracy and reduces memory usage by 95.62%, thus obtaining a low-cost model for semantic segmentation of extensive areas with a delay of only 5 seconds.

Chapter 8

Conclusions

This paper aimed to develop a fully autonomous algorithm for land cover identification, applied for deforestation monitoring, which does not require human intervention. In this way, the method should be able to segment an aerial image with the best possible accuracy without the need for manual labeling of pixels or objects or any other external human interventions. The motivation behind choosing this application was that deforestation is a global problem, out of control and with serious consequences for the environment and everyone, and traditional monitoring through the presence of environmental activists or authorities on-site is costly, time-consuming, and inefficient.

8.2 Original contributions

The original contributions of the doctoral thesis are as follows:

- Study of Clustering Methods (C1, C2, C4, C5): The thesis investigated various clustering methods applied to multispectral aerial images, analyzing the influence of their specific parameters on classification results. It compared these methods in terms of allocated memory and computation time to determine the most performant method.
- Color Extraction Module (C3): A color extraction module was implemented to represent the clusters identified by unsupervised methods. While necessary for internal thesis needs, this module can be applied to any study using unlabeled data for result representation and better operator analysis.
- Drastic Reduction in GMM Prediction Time (J2): The thesis achieved a drastic reduction in the prediction time of the Gaussian Mixture Model (GMM) algorithm by implementing a methodology based on Discrete Cosine Transform (DCT) for multispectral image processing and extracting the most

important coefficients. Traditionally, GMM can have very slow responses, especially when there are too many clusters to identify or when the images are too large. Unlike supervised techniques, where you can implement the trained model wherever necessary for future predictions, in image clustering, data preprocessing, and centroid calculation are steps performed along with prediction, not before implementing the solution. Despite an increase in preprocessing time, the proposed methodology significantly reduced training and prediction time.

- Automatic Pixel Labeling (J1): The thesis implemented a completely unsupervised technique to label multispectral images without any other supervised training or pre-trained networks as a starting point. Compared to other studies, the dataset used consisted of proprietary aerial images recorded with modern capabilities. To demonstrate the technique's success, field reality and corresponding labels were tested against multiple U-Net architectures to prove its usefulness and effectiveness. The study demonstrated that the labels are precise enough for U-Net to learn from them and predict very good results without human intervention in manual labeling.
- Reduction of U-Net Model Complexity (J1): Most real-world implementations of AI models are carried out under challenging conditions or with very weak hardware capabilities, not to mention the need for data analysis and rapid prediction. The presented study conducted tests on simplified U-Net models called S-Unet and achieved similar precise results to the original architecture while reducing the model's complexity by up to 60 times. The complexity reduction also decreased the average computation time and system memory allocation, resulting in significant cost reduction.
- Fast Segmentation of Extensive Areas (J2): Satellite and aerial images contain a large number of pixels due to their wide field of view (FOV) and high resolution. This makes segmenting these images much slower than everyday images containing objects. The proposed methods can predict thousands of square kilometers of multispectral images in a reasonable time frame.
- Development of a Fast, Low-Cost, and Environmentally Friendly Deforestation Monitoring Algorithm (J1): For example, S-Unet8 requires 28.5 Mb to predict 106 pixels in about 5 seconds. This algorithm can substantially aid in combating illegal deforestation. By implementing it in a real environment, a workflow is created for monitoring areas of interest, identifying offenders, and enforcing laws.

8.3 List of original publications

8.3.1 Articles published in international scientific journals

(J1) **A.-T. Andrei**, O. Grigore, „*Low-Cost Optimized U-Net Model with GMM Automatic Labeling Used in Forest Semantic Segmentation*”, Sensors, 23(21), p. 8991, 2023, **WOS:001099507300001**, Q2 article, Impact factor: 3.9, eISSN:1424-8220, DOI:10.3390/s2321899

(J2) **A.-T. Andrei**, O. Grigore, „*Development of a Very Low-Cost Deforestation Monitoring System Based on Aerial Image Clustering and Compression Techniques*", Advances in Electrical and Computer Engineering, 24(2), pp. 73-84, 2024, **WOS:001242091800008**, Q4 article, Impact factor: 0.8, eISSN: 1844-7600, DOI:10.4316/AECE.2024.02008

(J3) **A.-T. Andrei**, O. Grigore, „*Study of Clustering Algorithms for the Development of a Fast Deforestation Monitoring Tool*", Forests, Q1 article, Impact factor: 2.7, eISSN: 1999-4907 - sent to publication

8.3.2 Papers in international scientific conference proceedings indexed by WOS

(C1) **A.-T. Andrei**, O. Grigore, „*Unsupervised Machine Learning Algorithms Used in Deforested Areas Monitoring*", 2021 International Conference on e-Health and Bioengineering (EHB), pp. 1-4, Iași, România, 2021, **WOS:000802227900197**, ISSN:2575-5145, ISBN:978-1-6654-4000-4, DOI:10.1109/EHB52898.2021.9657737.

8.3.3 Papers in international scientific conference proceedings indexed by international databases

(C2) **A.-T. Andrei**, O. Grigore, „*Mean Shift Clustering with Bandwidth Estimation and Color Extraction Module Used in Forest Segmentation*", 2023 13th International Symposium on Advanced Topics in Electrical Engineering (ATEE), pp. 1-6, București, Romania, 2023, ISSN:1843-8571, ISBN:978-1-4799-7514-3, DOI:10.1109/ATEE58038.2023.10108106

(C3) **A.-T. Andrei**, O. Grigore, „*Color Extraction Module for Unsupervised Image Classification Representation*", 2023 13th International Symposium on Advanced Topics in Electrical Engineering (ATEE), pp. 1-5, București, Romania, 2023, ISSN:1843-8571, ISBN:978-1-4799-7514-3, DOI:10.1109/ATEE58038.2023.10108115

(C4) **A.-T. Andrei**, O. Grigore, „*Gaussian Mixture Model Application in Deforestation Monitoring*", 2022 International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), pp. 26-31, Ankara, Turcia, 2022, ISSN:2770-7962, ISBN:978-1-6654-7013-1, DOI:10.1109/ISMSIT56059.2022.9932845

(C5) **A.-T. Andrei**, O. Grigore, „*Combating Deforestation Using Different AGNES Approaches*", 2022 14th International Conference on Communications (COMM), pp. 1-5, București, România, 2022, ISBN:978-1-6654-9485-4, DOI:10.1109/COMM54429.2022.9817217

8.3.4 Scientific research reports

(R1) **A.-T. Andrei**, Coordinator: O. Grigore, „*Study and preparation of multispectral aerial data"*

(R2) **A.-T. Andrei**, Coordinator: O. Grigore, „*Algorithms applied to multispectral images"*

(R3) **A.-T. Andrei**, Coordinator: O. Grigore, „*Study of supervised learning methods"*

(R4) **A.-T. Andrei**, Coordinator: O. Grigore, „*Study of unsupervised learning methods"*

Bibliography

[6] S. Bhagwat, *The History of Deforestation and Forest Fragmentation: A Global Perspective*, Global Forest Fragmentation, pp. 5-19, 2014.

[7] J. Shroder, R. Sivanpillai, *Biological and Environmental Hazards, Risks, and Disasters*, Elsevier, pp. 313-315, 2015.

[8] A. Angelsen, D. Kaimowitz, *Rethinking the Causes of Deforestation: Lessons from Economic Models*, The World Bank Research Observer, 14(1), p. 73, 1999.

[10] B. Usman, *Satellite Imagery Land Cover Classification using K-Means Clustering Algorithm Computer Vision for Environmental Information Extraction*, Elixir Comp. Sci. & Engg., 2013.

[11] X. Zheng, Q. Lei, R. Yao, Y. Gong, Q. Yin, *Image segmentation based on adaptive K-means algorithm*, EURASIP Journal on Image and Video Processing, 2018.

[13] K. Pearson, *Contributions to the Mathematical Theory of Evolution*, University College, London, 1894.

[14] A. P. Dempster, N. M. Laird, D. B. Rubin, *Maximum Likelihood from Incomplete Data via the EM Algorithm*, Royal Statistical Society, 1977.

[15] Y. Tarabalka, J. A. Benediktsson, J. Chanussot, *Spectral–Spatial Classification of Hyperspectral Imagery Based on Partitional Clustering Techniques*, IEEE Transactions on Geoscience and Remote Sensing, 47(8), pp. 2973-2987, 2009.

[16] H. Permuter, J. Francos, I. Jermyn, *A study of Gaussian mixture models of color and texture features for image classification and segmentation,* Pattern Recognition, 39(4), pp. 695-706, 2006.

[17] R. Farnoosh, B. Zarpak, *Image Segmentation Using Gaussian Mixture Model*, IUST International Journal of Engineering Science, 19(1-2), pp. 29-32, 2008.

[21] Y. LeCun et al., *Backpropagation Applied to Handwritten Zip Code Recognition*, Neural Computation, 1(4), pp. 541-551, 1989.

[22] A. Krizhevsky, I. Sutskever, G.E. Hinton, *ImageNet classification with deep convolutional neural networks*, Communications of the ACM, 60(6), pp. 84–90, 2012.

[25] O. Ronneberger, P. Fischer, T. Brox, *U-Net: Convolutional Networks for Biomedical Image Segmentation,* Medical Image Computing and Computer-Assisted Intervention, Springer, 9351, pp. 234–241, 2015.

[41] J. Zhang et al., *LCU-Net: A novel low-cost U-Net for environmental microorganism image segmentation*, Pattern Recognition, Springer, 115(4), 2021.

[43] L. Haoran, S. Yifei, T. Jun, X. Shengzhou, *Half-UNet: A Simplified U-Net Architecture for Medical Image Segmentation*, Frontiers in Neuroinformatics, 16, 2022.

[67] D. Comaniciu, P. Meer, *Mean shift: a robust approach toward feature space analysis*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(5), pp. 603-619, 2002.

[82] J. E. Cavanaugh, A. A. Neath, *The Akaike information criterion: Background, derivation, properties, application, interpretation, and refinements*, WIREs Computational Statistics, 11(3), p. 1460, 2019.

[86] M. Yaqub et al., *State-of-the-Art CNN Optimizer for Brain Tumor Segmentation in Magnetic Resonance Images,* Brain Sci, 10(7), p. 427, 2020.