Land cover classification from satellite imagery using deep learning algorithms

Valentina Moreno Ramírez
Dept. of Mathematical Sciences
EAFIT
Medellin, Colombia
vmorenor@eafit.edu.co

Isabella Montoya Henao
Dept. of Mathematical Sciences
EAFIT
Medellin, Colombia
imontoyah@eafit.edu.co

Juan Carlos Duque
Dept. of Mathematical Sciences
EAFIT
Medellin, Colombia
jduquec1@eafit.edu.co

***Abstract*—Land cover classification is an essential task for various applications such as land-use planning, environmental monitoring, and disaster management. Traditional methods of land cover classification rely on manual interpretation of satellite imagery, which is a time-consuming and tedious process. Deep learning algorithms have shown great potential in automating this process by learning complex representations of the satellite imagery. This paper provides a comprehensive review of a deep learning algorithm used for land cover classification. The paper also discusses the challenges and future research directions in this field.**

***Index Terms*—Land cover, land use, Deep Learning, convolutional neural networks, improve, classification.**

I. Introduction

In recent years, the development of various remote sensing techniques and the availability of satellite-based geographic data have led to numerous applications in the domains of agriculture, disaster recovery, climate change, urban development, and environmental monitoring. The implementation of these strategies in numerous industrial processes across diverse sectors has had significant economic and environmental impacts. One of the most crucial techniques in remote sensing is the classification of land cover in satellite images, which refers to the process of identifying and categorizing different types of land surfaces present in a satellite image, such as forests, bodies of water, urban areas, and so on. This process involves the identification and separation of various types of land cover present in an image based on the analysis of its spectral properties, such as reflectance and texture, in order to classify the different characteristics of the Earth's surface into distinct categories Hu *et al.* (2021). The process can be carried out automatically using computational methods, such as the application of deep learning algorithms, which enable faster and more accurate land cover classification compared to traditional methods. The critical assessment of these techniques is essential for ensuring the most effective and appropriate use of remote sensing data, ultimately contributing to more sustainable and informed decision-making processes in various sectors.

Conventional image feature extraction and image analysis techniques often employ common computer vision methods, such as scale-invariant feature transformation Kobayashi (2014), oriented gradient histograms Negrel *et al.* (2014), and the bag of visual words (BoVW). All these approaches have been successful to varying degrees in a wide range of image classification modalities, including regular photography, medical imaging, and remote sensing, to name just a few. However, these techniques may not be sufficiently accurate in identifying objects in complex situations or with overlapping or masked objects, in addition to representing a costly process in terms of time and computational resources that often requires fine-tuning of parameters to achieve good results in different applications and datasets. Nevertheless, with the advent of convolutional neural networks (CNN) and deep learning in the 1980s, a new and promising research avenue has emerged. Unlike conventional techniques, CNNs can learn task-specific features from training data, allowing them to capture complex and discriminative patterns from images, greatly contributing to achieving our objectives.

The application of convolutional neural networks (CNNs) in land cover classification can make substantial contributions to geospatial data analysis. In urban planning, CNN-based land cover classification enables the identification of urban areas, green spaces, and water bodies. This information empowers urban planners to make informed decisions regarding the optimal placement and design of infrastructure, ensuring sustainable and well-designed urban environments (Hu *et al.*, 2021). In precision agriculture, CNNs play a vital role by accurately identifying areas with varying vegetation and soil conditions. This data allows farmers to adapt their agricultural practices, optimizing irrigation, fertilization, and other interventions to improve crop productivity and resource efficiency (Dari *et al.*, 2020). In natural resource management, CNNs have proven invaluable for land cover classification. They enable the identification and monitoring of protected areas, facilitate forest management planning, and detect changes in land cover over time. These advancements support effective conservation efforts and sustainable resource management practices (Lu *et al.*, 2004). Additionally, in environmental impact assessment, CNN-based land cover classification serves as a powerful tool for evaluating the environmental effects of development projects. It aids in identifying critical conservation areas and assessing the potential impacts of human activities on the natural environment, promoting informed decision-making and balancing development with environmental preservation (Chakroun, 2017). Overall, the use of CNNs in land cover classification has significantly advanced geospatial data analysis, providing accurate and detailed land cover mapping for informed decision-making, improved resource management.

A correct land cover classification is a fundamental task in the field of remote sensing and environmental science, and the use of convolutional neural networks (CNN) has represented a significant breakthrough in this area. CNNs are capable of automatically learning and extracting features from satellite images, making them particularly suitable for land cover classification without the need for prior manual feature selection. Furthermore, CNNs can process large amounts of data and detect complex and subtle patterns in images. These characteristics allow for more accurate and detailed land cover classifications, which are essential for effective natural resource management and land-use planning. In the following sections, we will present the workings of these networks in greater detail and the implementation of a competent CNN for land cover classification.

II. State of art

In the early stages of 1971, various governmental agencies recognized the increasing necessity for a comprehensive understanding of national land use and land cover patterns, and their associated environmental implications. This urgent need prompted the initiation of innovative land use and land cover classification models that could effectively integrate data from both conventional sources and advanced remote sensors mounted on high-altitude aircraft and satellite platforms (Anderson *et al.*, 1976). Nevertheless, the scarcity of available data at the time posed significant challenges in advancing the field. As a result, critical evaluation of the models' accuracy and effectiveness became increasingly important, highlighting the need for robust data collection methods and ongoing refinement of the classification algorithms.

In light of this, the development and refinement of land use and land cover classification models have proven to be instrumental in facilitating well-informed decision-making processes for land management and environmental policy. Moreover, these models have became a crucial aspect of sustainability and regional development, as emphasized by Basheer *et al.* (2022). These models are considered fundamental in the field of remote sensing (RS) and geographic information systems (GIS), automating the extraction of crucial information from the Earth's surface. In this way, detailed LULC information over large areas is essential for a wide range of urban and natural resource management issues (Ali & Johnson, 2022), such as change detection, environmental monitoring, urban expansion control, infrastructure planning, and biodiversity analysis (Carranza-García *et al.*, 2019).

Recent advances in machine learning and deep learning techniques have revolutionized the field of LULC classification models. Among these techniques, Convolutional Neural Networks (CNNs) have emerged as the state-of-the-art method for image classification in computer vision and machine learning, demonstrating impressive results on image classification challenges (Helber *et al.*, 2017). In particular, CNNs have been successfully applied to the analysis of remote sensing imagery, allowing for more accurate and efficient land use and land cover classification. Moreover, a study conducted by Basu et al. examined deep belief networks, basic CNNs, and stacked denoising autoencoders on the SAT-6 dataset (Ali & Johnson, 2022). The results of the study demonstrated that CNNs significantly outperformed the state-of-the-art by approximately 11% and 15% on the SAT-4 and SAT-6 datasets, respectively. These findings provide evidence of the potential of CNNs for LULC classification and highlight their superiority over other deep learning techniques in certain contexts.

Other machine learning techniques such as Random Forest, Support Vector Machines (SVMs), and Artificial Neural Networks (ANNs) have also been extensively used for LULC classification. These advancements in machine learning have undoubtedly opened up new possibilities for more accurate and efficient land use and land cover classification, which can greatly aid in the planning and management of natural resources and ecosystems. As cited in Carranza-García *et al.* (2019), a framework was developed for evaluating LULC classification using convolutional neural networks, support vector machines, random forests, and k-nearest-neighbors. The study found that deep learning represents a potent solution for LULC classification, as it yielded promising results for all images studied and was reported to be the fastest for both training and testing (Carranza-García *et al.*, 2019). In contrast, Basheer *et al.* (2022) compared the results of LULC classifiers using different satellite imagery and three machine learning techniques, including support vector machines (SVM), maximum likelihood (ML), and Random Forest (RF). Their findings indicated that SVM achieved the best results for the various testing data (Basheer *et al.*, 2022). This highlights the importance of carefully selecting and tailoring machine learning techniques to the specific requirements and contexts of LULC classification tasks.

III. Materials and Methods

A. Description of Datasets

The process of data collection for the training of supervised algorithms can often be complex and laborious, requiring high-quality, high-resolution images depicting the Earth's surface. Frequently, these images are sourced from satellites and drones, access to which can be expensive. Furthermore, once the images have been obtained, each one must be accurately classified with detailed information regarding the Earth's surface characteristics, such as the presence of water, vegetation, buildings, roads, and so forth. This process can be particularly tedious and time-consuming, as it necessitates manual classification for each image.

Presently, we find ourselves amidst a transition towards the public and continual availability of satellite image data for Earth observation. Pertinent to this, the European Space Agency's Copernicus program and the National Aeronautics and Space Administration's Landsat program are spearheading significant initiatives aimed at making this data freely accessible for both commercial and non-commercial use. The goal is to foster innovation and entrepreneurship. The availability of this data allows for a broad range of applications, including but not limited to agriculture, disaster recovery, climate change, urban development, and environmental monitoring (Bischke *et al.*, 2017).

Various classified datasets of land use can be found, one of which is the UC Merced (UCM) dataset, introduced by Yang & Newsam (2010). This dataset comprises a total of 21 categories of land use and coverage. The images measure 256x256 pixels with a high spatial resolution of approximately 30 cm per pixel and are in RGB color space. Despite these attributes, the dataset consists of only 100 images per category. Consequently, when implementing our convolutional neural network, the training of the network could lead to overfitting. That is, the network becomes excessively adjusted to the training data and loses its capacity to generalize and classify new images, unable to learn varied patterns and complex relationships.

A research team from the Technical University of Munich, in collaboration with the European Space Agency (ESA), created EuroSAT, a high-resolution satellite image dataset used for land coverage classification in Europe. This dataset consists of 27,000 satellite images taken by ESA's Sentinel-2 satellite, boasting a spatial resolution of 10 meters per pixel. The images were captured across 13 different spectral bands, spanning from ultraviolet to near-infrared, enabling detailed classification of land coverage and high precision in detecting changes and variations on the Earth's surface. It includes ten different land coverage classes, each with approximately 2,700 images, making this dataset ideal for training and validating the convolutional neural network to be implemented. The land coverage classes include industrial buildings, residential buildings, sea lake, herbaceous vegetation, annual crop, highway, permanent crop, pasture, river, and forest. It should be noted for proposed analyses that this proposed dataset has not undergone atmospheric correction, which may result in images with a color cast which may affect the accuracy and quality of the image. Consequently, CNN can learn wrong patterns and produce inaccurate results when classifying new images.

Here are some images of the EUROSAT dataset with their respective category:

Fig. 1: Annual Crop

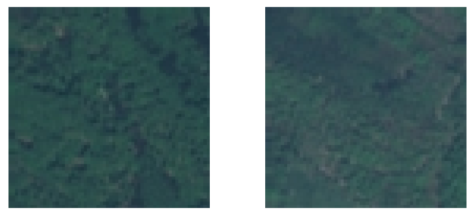

Fig. 2: Forest

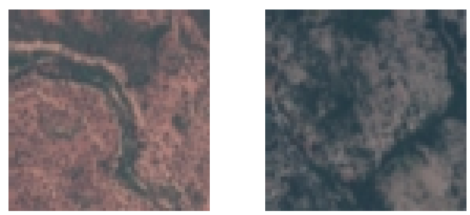

Fig. 3: Herbaceous Vegetation

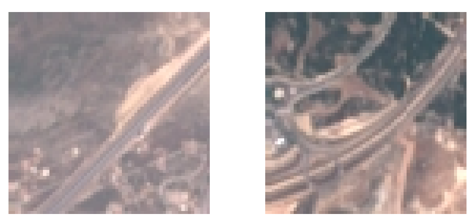

Fig. 4: Highway

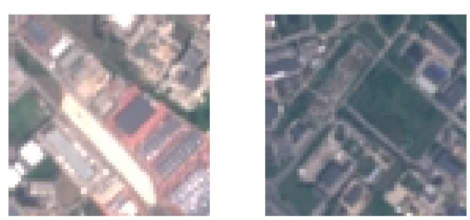

Fig. 5: Industrial

Fig. 6: Pasture

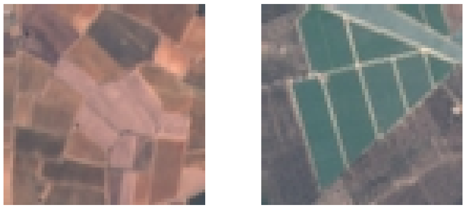

Fig. 7: Permanent crop

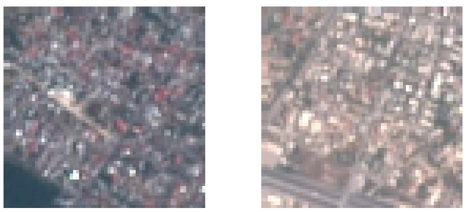

Fig. 8: Residential

Fig. 9: River

Fig. 10: Sea-Lake

B. Convolutional Neural Networks

Object identification in images is one of the classic problems in Artificial Intelligence. In the early days of image identification technology, machine learning relied on neural networks and was more than adequate for detecting elements in small images. However, these same algorithms become inefficient when we increase the size of such images. As the image size increases, the number of pixels produces an exponential increase in input variables (features) that are impossible to handle by a traditional neural network architecture.

Convolutional neural networks (CNNs) solve this problem as they assume certain spatial characteristics of the inputs, allowing for a simplification of the network architecture and significantly reducing the number of input variables. As a result, they are especially useful in computer vision problems, and in particular, object recognition.

A CNN typically takes a rank-3 tensor as input, in our case, an image with 256 rows, 256 columns, and 3 channels (color channels R, G, B). The input then passes sequentially through a series of processing steps. A processing step is generally called a layer, which could be a convolutional layer, pooling layer, normalization layer, fully connected layer, loss layer, etc.

At the beginning of the network lies the feature extraction phase, composed of convolutional neurons and subsampling reduction. At the end of the network are simple perceptron neurons to perform the final classification on the extracted features. The feature extraction phase resembles the stimulating process in visual cortex cells. This phase is composed of alternating layers of convolutional neurons and subsampling reduction neurons. As the data progresses through this phase, its dimensionality decreases, with neurons in distant layers being much less sensitive to disturbances in the input data, while being activated by increasingly complex features.

Convolutional neurons: In the feature extraction phase, the simple neurons of a perceptron are replaced by matrix processors that perform an operation on the 2D image data that passes through them, instead of a single numerical value. The output of each convolutional neuron is calculated as follows:

$$Y_j = g\left(b_j + \sum_i K_{ij} \otimes Y_i\right)$$

Where the output Y_j of a neuron j is a matrix that is calculated by means of the linear combination of the outputs Y_i of the neurons in the previous layer, each of them operated with the corresponding convolutional kernel K_{ij} for that connection. This amount is added to an influence b_j and then passed through a non-linear activation function $g(\cdot)$.

The convolution operator has the effect of filtering the input image with a previously trained kernel. This transforms the data in such a way that certain features (determined by the shape of the kernel) become more dominant in the output image, as they are assigned a higher numerical value to the pixels that represent them. These kernels have specific image processing abilities, such as edge detection, which can be performed with kernels that highlight the gradient in a particular direction. However, kernels trained by a convolutional neural network are generally more complex in order to extract other more abstract and non-trivial features.

Downsampling Neurons: Neural networks have a certain tolerance to small perturbations in the input data. For example, if two nearly identical images (distinguished only by a shift of a few pixels laterally) are analyzed with a neural network, the result should be essentially the same. This is achieved, in part, due to the downsampling that occurs within a convolutional neural network. By reducing the resolution, the same features will correspond to a larger activation field in the input image.

The pooling layer (POOL) is a type of layer that is present in a large number of CNN architectures. Its usefulness lies in reducing the representations obtained so that they become smaller and more computationally manageable, reducing the number of necessary parameters. The max-pooling operation finds the maximum value among a sample window and passes this value as a summary of characteristics over that area. As a result, the size of the data is reduced by a factor equal to the size of the sample window on which it operates.

Classification Neurons: After one or more feature extraction phases, the data finally reaches the classification phase. By then, the data has been refined to a set of unique features for the input image, and it is now the task of this final phase to classify these features into one label or another, according to the training objectives. These neurons are found on the fully connected layers that are the responsible for the final classification of the extracted features, generating an output that corresponds to the desired classes or categories

$$y_j = g\left(b_j + \sum_i w_{ij} \cdot y_i\right)$$

Where the output y_j of neuron j is a value calculated by the linear combination of the outputs y_i from the neurons in the previous layer, each multiplied by a weight w_{ij} corresponding to that connection. This quantity is then added to an influence b_j and passed through a non-linear activation function $g(\cdot)$.

C. CNN Architecture

For this project, two different variants of convolutional neural networks have been developed. This approach has been adopted with the intention of determining which of these network architectures is more suitable for addressing the specific problem under study. Therefore, in the subsequent section, a detailed description is provided for two convolutional neural network models: the classic LeNet-5 and a network designed and implemented by the authors for this project.

Own Convolutional Neural Network

The implemented neural network contains 10 convolutional layers, 5 pooling layers and 2 fully connected layers. In addition, the categorical cross entropy and the ADAM optimizer, also known as Adaptive Moment Estimation, were used as the loss function. This is an optimization algorithm popularly used in the training of neural networks. It combines elements of the RMSprop and momentum algorithms to dynamically adapt the learning rates of model parameters during training. The Adam optimizer maintains an individual adaptive learning rate for each model parameter. In addition, it uses estimates of the first order moment (gradients) and of the second order moment (cumulative moment of the squares of the gradients) to calculate the updates of the weights.

In figure 11 are the specifications of each layer of the convolutional neural network.

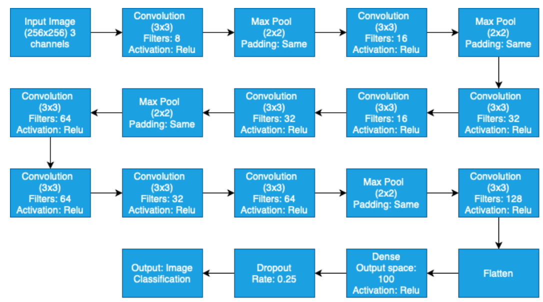

Fig. 11: Implemented convolutional neural network

CNN LeNet-5

LeNet-5 is one of the oldest and most well-known Convolutional Neural Networks (CNN) designed by Lecun *et al.* (1998) for handwritten and hand-printed character recognition. The LeNet-5 model is a self-learning process that includes integrated learning, backpropagation, and selection optimization, thus combining feature extraction and image recognition. This CNN consists of 7 layers, not counting the input layer. These layers include two convolutional layers, two pooling layers, two fully connected layers, and an output layer (Rosmala & Rifaldy, 2023). A schematic diagram of the LeNet-5 model's structure is shown in Figure 12.

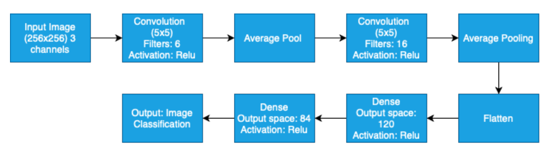

Fig. 12: LeNet-5 structure

Ultimately, LeNet-5 played a crucial role in the development of CNNs and was one of the pioneers on the path towards more complex and deeper architectures that are used today. Its focus on feature extraction and its ability to learn visual representations have influenced the design of many modern convolutional neural networks, enabling significant advancements in tasks such as object recognition, semantic segmentation, face detection, and numerous other computer vision applications (Wang & Raj, 2015).

D. Metrics to evaluate the model

In classification models, there are various metrics that help assess the performance of the algorithm, each focusing on different aspects depending on the parameter being analyzed. Some of these metrics will be presented below, explaining how they are calculated and what they represent in the model.

1) Confusion Matrix: The confusion matrix is a tool in the field of artificial intelligence and machine learning that allows for visualization of algorithm performance (Aniruddha, 2020). It presents the following possibilities:

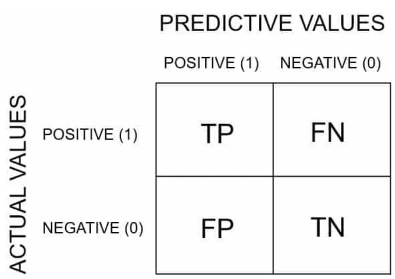

Fig. 13: Confusion matrix

Where TP are true positives, FN false negatives, FP false positives and TN true negatives.

As can be seen, this matrix has the actual values in its rows and the prediction results in its columns. If both the actual and predicted values are true or false, it indicates that the model classified correctly. For this work, this occurs when a person with heart disease is predicted to be ill, or when a person is determined to be healthy when they actually are. These two options are desired, but this does not always occur.

On the one hand, the model may predict that a certain sample meets a particular condition when it does not, which is the case for a false positive, also known as a type I error. On the other hand, when the model predicts false when the answer is true, there is a false negative, or type II error. Specifically, for heart diseases, the type I error occurs when a certain individual is said to be ill when they are actually healthy, and the type II error occurs when a person is predicted to be healthy but actually has some heart disease.

Therefore, the confusion matrix will contain the frequency of each of these four possibilities, for which it is expected that false positives and false negatives are much lower than true positives and true negatives. This matrix allows for the calculation of various metrics that will be presented below:

- **Accuracy:** This calculates how close the result of a measurement is to its true value, which, in statistical terms, relates to the bias of the estimate. It is calculated using the following formula:

$$\text{Accuracy} = \frac{\text{True positives} + \text{True negatives}}{\text{Total predictions}}$$

  In other words, accuracy calculates the proportion of correct predictions made (Barrios, 2019). This is not a good metric when working with imbalanced datasets, as it can yield very good results when the model is very poor or vice versa (N.A, 2019).

- **Precision:** This refers to the dispersion of the set of obtained results; if there is less dispersion, there will be greater precision. It is calculated as follows:

$$\text{Precision} = \frac{\text{True positives}}{\text{True positives} + \text{False positives}}$$

  This represents the percentage of positive cases detected by the model (Barrios, 2019).

- **Recall:** This indicates the proportion of true positives that were correctly estimated by the algorithm, calculated as follows:

$$\text{Recall} = \frac{\text{True positives}}{\text{True positives} + \text{False negatives}}$$

  This indicates the algorithm's ability to correctly predict true cases and is also known as the true positive rate (Barrios, 2019).

Since all these measures are proportions, they always take values in the range [0,1], and the closer they are to 1, the better the model is in this aspect (Barrios, 2019).

2) F1 score: The F1 score is an excellent metric when working with imbalanced data, as is the case in this work, because it combines precision and recall into a single metric. It is calculated as the harmonic mean of precision and recall, that is:

$$\text{F1 score} = 2 * \frac{\text{PrecisionRecall}}{\text{Precision} + \text{Recall}}$$

Since both precision and recall take values between 0 and 1, this metric will also do so, reaching its best value at the upper end of this range (Korstanje, 2021).

IV. Results and Discussion

In this section, we present and discuss the results obtained from the testing experiments and the subsequent statistical analysis. Finally, we also show the accuracies and classification maps obtained for both neuronal networks.

A. Proposed model

In order to understand how the implemented convolutional neural network (CNN) is learning and adjusting over time, we refer to Figure 14, which compares epochs versus accuracy, and Figure 15 that presents epochs versus losses.

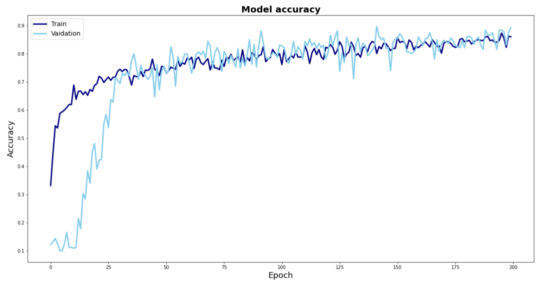

Fig. 14: Model accuracy

From Figure 14, an increasing trend can be observed, suggesting that the neural network is learning and improving its accuracy with each epoch. Furthermore, two curves are visible representing the training dataset (in dark blue) and the validation dataset (in light blue). The similar upward trend in these curves leads to the conclusion that the model is not overfitting. Finally, around the 150-epoch mark, the model starts to stabilize with an accuracy of approximately 0.86.

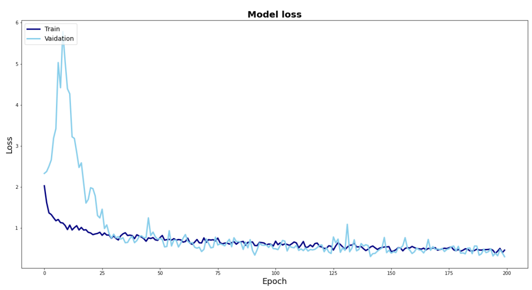

Fig. 15: Model loss

From Figure 15, it can be seen that both the training and validation data show a decreasing trend, indicating that the network is learning and reducing its error with each epoch. Furthermore, as both curves follow a similar downward trend, it can be confirmed that the model is not experiencing overfitting. Ultimately, after about 150 epochs, the training and validation curves approach zero, indicating optimal model performance.
On the other hand, to determine the performance and efficacy of the CNN, several metrics such as recall, precision, accuracy, and F1 score were measured.

Class	Recall	Precision	Accuracy	F1 Score
AnnualCrop	0.8	0.9	0.97	0.85
Forest	0.97	0.94	0.99	0.96
HerbaceousVegetation	0.72	0.84	0.95	0.77
Highway	0.64	0.85	0.96	0.73
Industrial	0.97	0.9	0.99	0.93
Pasture	0.85	0.79	0.97	0.82
PermanentCrop	0.82	0.72	0.95	0.76
Residential	0.98	0.95	0.99	0.96
River	0.88	0.68	0.95	0.77
SeaLake	0.92	1.0	0.99	0.95

Accuracy: 85.59

Fig. 16: Evaluation metrics

The implemented convolutional neural network has shown promising performance, achieving an accuracy of 85.59% in the classification of land use and land cover images for 10 different categories. In most cases, the recall value exceeds 0.8, indicating that the network has correctly identified over 80% of the images in each category, demonstrating a notable ability to detect relevant features.

However, it can be observed that the precision in some classes, such as 'River,' 'Permanent crop,' and 'Pasture,' is relatively low, indicating that the network may be confusing these classes with others, possibly due to their similarities. Despite this, in the rest of the classes, precision exceeds 84%, indicating generally solid performance.

On the other hand, the accuracy measure exhibits excellent results, exceeding 95% for all classes. This suggests that in most cases, the network assigns the correct class to the images. However, it is important to consider that this measure may be less reliable if the classes are imbalanced, which could artificially inflate the accuracy.

Finally, the F1 score, a metric that combines precision and recall and is particularly useful for evaluating a model's effectiveness in imbalanced class situations, was evaluated. Although the results are not exceptional, they are still satisfactory, with F1 score values ranging from 73% to 95%. These values indicate that the convolutional neural network maintains a balance between precision and recall, supporting overall solid and reliable performance.

Lastly, to further verify the efficiency of the implemented convolutional neural network, a confusion matrix was generated, as shown in Figure 17.

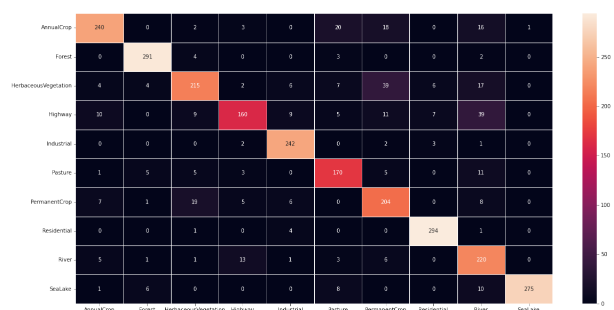

Fig. 17: Confusion matrix

In the confusion matrix, each row corresponds to instances of a real class, while each column reflects instances of a predicted class. The diagonal components represent correct predictions, and the off-diagonal elements reveal instances where images from one class were misclassified as another.
The colors on the diagonal of the confusion matrix indicate the number of correctly predicted images, becoming lighter as the precision increases. However, these values can be biased due to class imbalance, which may lead to the interpretation

of a lower number of correctly predicted images in certain classes.

In this context, the classes that the neural network predicts with the highest precision are *Forest*, *Industrial*, and *Residential*, with only 9, 8, and 6 incorrectly classified images, respectively. On the other hand, the *Herbaceous Vegetation* and *Highway* classes showed the highest classification errors. The *Herbaceous Vegetation* class has a total of 85 incorrectly classified images, most frequently confused with the *Residential* class. In the case of the *Highway* class, there were 90 misclassified images, with the most frequent confusion occurring with the *River* class.

In conclusion, the results indicate that the CNN demonstrates considerable efficiency and accuracy in classifying the 10 land use classes. This analysis allows us to conclude that the CNN offers a solid and efficient performance, laying a strong foundation for future improvements and optimizations.

B. LeNet-5

The same previous tests were performed for the LeNet-5 model to determine how the network was learning and adjusting over time, in order to compare the behavior of both models.

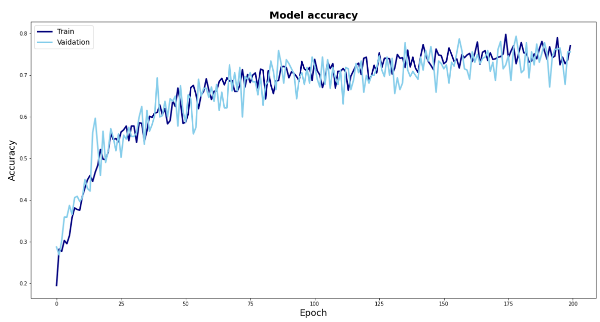

Fig. 18: LeNet-5 accuracy

From Figure 18, it can be observed that both the training and validation data show an increasing trend, indicating that the neural network is improving its accuracy with each epoch. Additionally, it can be concluded that the model does not exhibit overfitting. Furthermore, it can be noticed that around epoch 175, the model starts to stabilize with an accuracy of approximately 0.75.

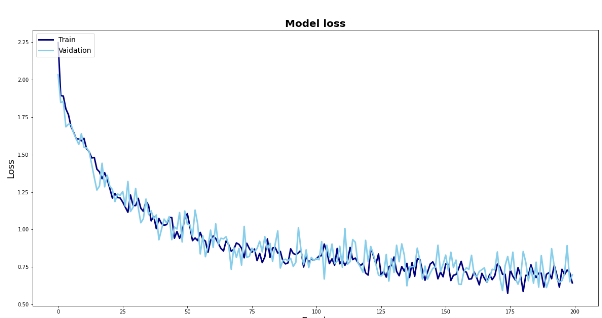

Fig. 19: LeNet-5 Loss

From Figure 19, it can be observed that both the training and validation curves exhibit similar behavior, which again confirms that the model is not experiencing overfitting. Additionally, it can be seen that the curves progressively decrease and eventually, around epoch 175, they start to stabilize at an error rate of approximately 0.5.

Class	Recall	Precision	Accuracy	F1 Score
AnnualCrop	0.85	0.81	0.96	0.83
Forest	0.97	0.8	0.97	0.88
HerbaceousVegetation	0.68	0.7	0.93	0.69
Highway	0.4	0.5	0.91	0.44
Industrial	0.91	0.85	0.98	0.88
Pasture	0.64	0.77	0.96	0.7
PermanentCrop	0.44	0.75	0.93	0.55
Residential	0.98	0.73	0.96	0.84
River	0.7	0.66	0.94	0.68
SeaLake	0.89	0.94	0.98	0.91

Accuracy: 75.89

Fig. 20: LeNet-5 valuation metrics

The LeNet-5 model achieves an accuracy of 75.89% in classifying images belonging to ten different land use categories. It is noteworthy that the classes *Highway* and *Permanent Crop* exhibit relatively poor performance in terms of recall, with values of 0.4 and 0.44, respectively. Additionally, only half of the categories achieve a recall higher than 0.85, implying that the network is not effectively identifying images in a significant portion of the categories.

Regarding precision, the results are not particularly remarkable, as most classes obtain precision values ranging from 0.6 to 0.8. Only two classes fall outside this range: *Highway* with a precision of 0.5, and *SeaLake*, which achieves the highest precision with 0.94.

Although the *accuracy* measure offers seemingly promising results, surpassing 91% in all classes, this metric may not accurately reflect the model's performance due to class imbalance.

Next, the F1 score was calculated, a relevant metric when working with imbalanced classes. Although the results are not outstanding, it is notable that half of the classes achieve F1 scores higher than 83%. However, the remaining classes have lower values, ranging from 0.44 to 0.7, indicating that the model is not adequately classifying a considerable number of classes.

Finally, to further verify the efficiency of the implemented convolutional neural network, a confusion matrix was generated, as shown in Figure 21.

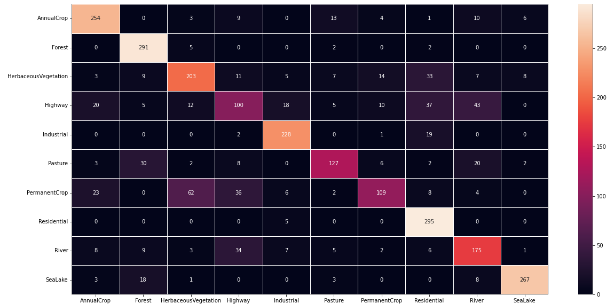

Fig. 21: LeNet-5 confusion matrix

For the confusion matrix of the LeNet-5 model, the classes that the neural network predicts with higher precision are *Forest* and *Residential*, with only 9 and 5 incorrectly classified images, respectively. On the other hand, the classes *Herbaceous vegetation*, *Highway*, and *Permanent crop* exhibited the highest classification errors.

The *Herbaceous vegetation* class has a total of 97 incorrectly classified images, and the class with which it had the most confusion was *Residential*. In the case of the *Highway* class, there were 150 incorrectly classified images, most frequently confused with the *Residential* and *River* classes. For the *Permanent crop* class, there were 141 incorrectly classified images, with *Herbaceous vegetation* being the most frequently confused class.

C. Comparison of Results

In summary, the results obtained by the implemented convolutional neural network significantly outperformed those of the LeNet-5 model. This improvement is mainly attributed to the greater number of layers in the implemented network, which allows for capturing more detailed image features. This ability to extract deeper and more detailed information from the images improves the classification accuracy and reduces confusion between different classes. Thus, the implemented convolutional neural network achieves a model accuracy of 85.59%, while the LeNet-5 model achieves an accuracy of 75.89%. Additionally, the following table shows the comparison of average values obtained for the different metrics.

	Own CNN	**LeNet-5**
Average Recall	0.855	0.746
Average Accuracy	0.857	0.751
Average Precision	0.971	0.952
Average F1 score	0.850	0.740

Based on the previous results, it can be concluded that the implemented convolutional neural network consistently offers superior performance. The implemented CNN outperformed the LeNet-5 model in terms of accuracy, precision, recall, and F1 score. This indicates that the developed CNN is more effective in classifying land use and land cover images using Sentinel-2 satellite data.

V. Conclusions

In this study, we addressed the problem of Land Use and Land Cover (LULC) classification using Sentinel-2 satellite imagery. The dataset consists of 27,000 labeled and georeferenced images spanning 10 classes and 13 different spectral bands. We evaluated the performance of two deep Convolutional Neural Networks (CNNs): the well-known LeNet-5 and a CNN developed by the authors.

Our results demonstrated that the implemented CNN outperformed the LeNet-5 model, achieving an accuracy of 85.59% compared to LeNet-5's 75.89%. This superiority also extended to precision, recall, and F1 score metrics, indicating a better ability to correctly classify the images and withstand confusion between classes.

However, we identified areas for improvement in both models, particularly in distinguishing the classes *Herbaceous vegetation*, *Highway*, and *Permanent crop*, which exhibited the highest classification errors. These areas could be addressed through gathering more training data, optimizing the network parameters, or developing additional features to help the network effectively differentiate between these categories.

Lastly, it is worth emphasizing that LULC classification is a field of increasing relevance, and the models used must become more accurate and efficient, especially in industrial contexts where accuracy can save time and resources. Thus, while the results of the implemented CNN are promising, we consider this model as a solid starting point for future improvements and optimizations, as the ultimate goal is to achieve a reliable and accurate classification for the different LULC classes.

Acnkowledge

Thanks to University EAFIT that allows mathematical engineering students to develop their interest in investigation. OCL to provide clear data of the world exports and imports and the World Bank, institution that developed the World Governance Indicators on which we based out method.

References

Ali, Kamran, & Johnson, Brian A. 2022. Land-Use and Land-Cover Classification in Semi-Arid Areas from Medium-Resolution Remote-Sensing Imagery: A Deep Learning Approach. *Sensors*, **22**(22).

Anderson, James R., Hardy, Ernest E., Roach, John T., & Witmer, Richard E. 1976. A Land Use and Land Cover Classification System for Use with Remote Sensor Data. *GEOLOGICAL SURVEY PROFESSIONAL*.

Aniruddha, Bhandari. 2020. Everything you Should Know about Confusion Matrix for Machine Learning. *Analytics Vidhya*.

Barrios, Juan. 2019. La matriz de confusión y sus métricas. *Health Big Data*.

Basheer, Sana, Wang, Xiuquan, Farooque, Aitazaz A., Nawaz, Rana Ali, Liu, Kai, Adekanmbi, Toyin, & Liu, Suqi. 2022. Comparison of Land Use Land Cover Classifiers Using Different Satellite Imagery and Machine Learning Techniques. *Remote Sensing*, **14**(19).

Bischke, Benjamin, Bhardwaj, Prakriti, Gautam, Aman, Helber, Patrick, Borth, Damian, & Dengel, Andreas. 2017. Detection of Flooding Events in Social Multimedia and Satellite Imagery using Deep Neural Networks. *In: MediaEval*.

Carranza-García, Manuel, García-Gutiérrez, Jorge, & Riquelme, José C. 2019. A Framework for Evaluating Land Use and Land Cover Classification Using Convolutional Neural Networks. *Remote Sensing*, **11**(3).

Chakroun, Hedia. 2017. Quality Assessment of MODIS Time Series Images and the Effect on Drought Monitoring. *Open Journal of Applied Sciences*, **07**(07), 365–383.

Dari, Jacopo, Brocca, Luca, Quintana-Seguí, Pere, Escorihuela, María José, Stefan, Vivien, & Morbidelli, Renato. 2020. Exploiting high-resolution remote sensing soil moisture to estimate irrigation water amounts over a Mediterranean region. *Remote Sensing*, **12**(16), 2593.

Helber, Patrick, Bischke, Benjamin, Dengel, Andreas, & Borth, Damian. 2017. EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification. 08.

Hu, Bin, Xu, Yongyang, Huang, Xiao, Cheng, Qimin, Ding, Qing, Bai, Linze, & Li, Yan. 2021. Improving urban land cover classification with combined use of sentinel-2 and sentinel-1 imagery. *ISPRS International Journal of Geo-Information*, **10**(8), 533.

Kobayashi, Takumi. 2014. Dirichlet-based histogram feature transform for image classification. *Pages 3278–3285 of: Proceedings of the IEEE conference on computer vision and pattern recognition*.

Korstanje, Joos. 2021. The F1 score.

Lecun, Yann, Bottou, León, Bengio, Yoshau, & Haffner, Patrick. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, **86**(11), 2278–2324.

Lu, Dengsheng, Mausel, Paul, Brondizio, Eduardo, & Moran, Emilio. 2004. Change detection techniques. *International journal of remote sensing*, **25**(12), 2365–2401.

N.A. 2019. Clasificación con datos desbalanceados. *Aprende Machine Learning*.

Negrel, Romain, Picard, David, & Gosselin, Philippe-Henri. 2014. Evaluation of second-order visual features for land-use classification. *Pages 1–5 of: 2014 12th International Workshop on Content-Based Multimedia Indexing (CBMI)*. IEEE.

Rosmala, Dewi, & Rifaldy, Muhamad Rizky. 2023. Implementation of architecture LeNet – 5 for Javanese scripts handwriting recognition. *AIP Conference Proceedings*, **2772**(1).

Wang, Haohan, & Raj, Bhiksha. 2015. A survey: Time travel in deep learning space: An introduction to deep learning models and how deep learning models evolved from the initial ideas. 10.

Yang, Yi, & Newsam, Shawn. 2010. Bag of visual words and spatial extensions for land-use classification. *Pages 270–279 of: Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*.