UNIVERSIDADE FEDERAL DO PARANÁ

MARIO ERNESTO JIJÓN PALMA

INTEGRATION OF STACKED-AUTOENCODERS AND CONVOLUTIONAL NEURAL NETWORKS FOR HYPERSPECTRAL IMAGE CLASSIFICATION

CURITIBA

2021

MARIO ERNESTO JIJÓN PALMA

INTEGRATION OF STACKED-AUTOENCODERS AND CONVOLUTIONAL NEURAL NETWORKS FOR HYPERSPECTRAL IMAGE CLASSIFICATION

Tese apresentada como requisito parcial à obtenção do título de Doutor em Ciências Geodésicas no Curso de Pós-Graduação em Ciências Geodésicas, Setor de Ciências da Terra, Universidade Federal do Paraná.

Orientador: Prof. Dr. Jorge Antônio Silva Centeno

CURITIBA

2021

Catalogação na Fonte: Sistema de Bibliotecas, UFPR
Biblioteca de Ciência e Tecnologia

P171i Jijón Palma, Mario Ernesto
Integration of stacked-autoencoders and convolutional neural networks for hyperspectral image classification [recurso eletrônico] / Mario Ernesto Jijón Palma. – Curitiba, 2021.

Tese - Universidade Federal do Paraná, Setor de Ciências da Terra, Programa de Pós-Graduação em Ciências Geodésicas, 2021.

Orientador: Jorge Antônio Silva Centeno.

1. Processamento de imagens. 2. Redes neurais (computação). I. Universidade Federal do Paraná. II. Centeno, Jorge Antônio Silva. III. Título.

CDD: 006.32

Bibliotecária: Vanusa Maciel CRB- 9/1928

MINISTÉRIO DA EDUCAÇÃO
SETOR DE CIENCIAS DA TERRA
UNIVERSIDADE FEDERAL DO PARANÁ
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO CIÊNCIAS GEODÉSICAS - 40001016002P6

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em CIÊNCIAS GEODÉSICAS da Universidade Federal do Paraná foram convocados para realizar a arguição da tese de Doutorado de **MARIO ERNESTO JIJÓN PALMA** intitulada: **INTEGRATION OF STACKED-AUTOENCODERS AND CONVOLUTIONAL NEURAL NETWORKS FOR HYPERSPECTRAL IMAGE CLASSIFICATION**, sob orientação do Prof. Dr. JORGE ANTONIO SILVA CENTENO, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de doutor está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 24 de Maio de 2021.

Assinatura Eletrônica
27/05/2021 18:21:42.0
JORGE ANTONIO SILVA CENTENO
Presidente da Banca Examinadora

Assinatura Eletrônica
31/05/2021 15:14:18.0
ANA PAULA DALLA CORTE
Avaliador Externo (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
01/06/2021 00:16:21.0
HIDEO ARAKI
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
27/05/2021 18:22:54.0
RODRIGO DE CAMPOS MACEDO
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
27/05/2021 18:29:46.0
GABRIEL HENRIQUE DE ALMEIDA PEREIRA
Avaliador Externo (SIMEPAR)

This thesis is dedicated with gratitude to:

My mother Clara Elena, my sister Fanny Elena, my princess Elena Sarahí, for being my support, strength and inspiration. I am proud of you and I will always admire you. I want to be with you all my life.

I also dedicate to Mari for being my support and because she is part of my life and happiness.

Mario Ernesto Jijón Palma
24/05/2021

ACKNOWLEDGMENTS

I would like to thank to my advisor, Prof. Dr. Jorge Antonio Silva Centeno, for their teachings, guides and orientations for the development of this research project that was successfully completed.

I would also like to thank to CAPES and CNPq/Brazil for the financial support.

To the professors and employees of the Postgraduate Course in Geodetic Sciences at the Federal University of Paraná.

To the secretary of the CPGCG, Mônica Kleuser, who was always willing to help me when I needed it.

To colleagues of the Remote Sensing Laboratory (LABSENSO)

To my family, especially to my aunt Inesita and grandmother Fanny, for the love they give me and because I am sure I can count on them for the rest of my life.

To my Ecuadorian friends and colleagues, for their personal support and shared moments, Lucia, Andrea, José Luis and Christain.

To my Brazilian friends and colleagues, for the welcome and for the moments shared during this doctoral process.

To the Mozambican community that supported with conversations and life learning, Caisse, Jaime, Andre and Miguel.

To Jaime Macuacua for his help in the discrimination of land cover classes for the elaboration of the thematic map of the Canguiri image.

Finally, I want to give special thanks to my mother, sister and niece Sarahí, for their unconditional love during the difficult times presented during these four years of my PhD, besides, they are my motivation to reach my goals and objectives.

To achieve your dreams you need discipline, hard work, effort and sacrifice, but above all you need to have the family as your source of inspiration.

RESUMO

Deep Learning ou aprendizado profundo abriu novas possibilidades para o pré-processamento, processamento e análise de dados hiperespectrais usando várias camadas de redes neurais e pode ser usado como ferramenta de extração de atributos. Nesta pesquisa, é desenvolvido um modelo híbrido baseado em pixels que integra *Stacked-Autoencoders* (SAE) y Redes Neurais Convolucionais (CNN) para classificar dados hiperespectrais. O núcleo do modelo integrado (SAE-1DCNN) é um *Autoencoder* que é aprimorado usando camadas convolucionais nas etapas de codificação (*encoding*) e decodificação (*decoding)*. Isso permite melhorar a discriminação de dados no treinamento não supervisionado e reduzir o tempo no processamento, pois permite uma descrição dos atributos baseada na assinatura hiperespectral do pixel e aproveita a eficácia da arquitetura profunda com base nas camadas convolucionais e pooling. Como filtros unidimensionais foram aplicados no modelo integrado, o tempo de processamento é consideravelmente menor do que ao usar filtros 2D-CNN. Em uma primeira etapa, o modelo SAE-1DCNN é usado para extração de atributos e, em seguida, esses resultados são usados em uma etapa final para uma classificação supervisionada. Assim, na primeira etapa os parâmetros da rede são ajustados usando amostras de treinamento e após na segunda etapa uma abordagem *fine-tuning* composta de regressão logística com base na função de ativação softmax foi aplicada para classificação. Três aspectos são analisados nesta pesquisa: a capacidade do modelo de excluir bandas ruidosas, sua capacidade de redução da dimensionalidade e seu potencial para realizar a classificação da cobertura da terra usando dados hiperespectrais. Os experimentos foram realizados com diferentes conjuntos de dados hiperespectrais: Indian Pines, Universidade de Pavia e Salinas, amplamente utilizados pela comunidade científica, e uma imagem hiperespectral capturada na Fazenda Canguiri da Universidade Federal do Paraná (UFPR) no Paraná-Brasil. Para validar a metodologia proposta, os resultados obtidos foram comparados aos métodos tradicionais de aprendizado de máquina para verificar o potencial da integração de autoencoders (AE) e redes convolucionais. Os resultados obtidos mostraram similaridade com os métodos tradicionais em termos de acurácia da classificação hiperespectral, porém demandaram menos tempo de processamento, portanto, a metodologia proposta (SAE-1DCNN) é considerada promissora, sólida e pode ser uma alternativa para o pré-processamento de dados hiperespectrais. e processamento.

Palavras-chaves: *deep learning; convolutional neural network* (CNN); *stacked-autoencoders* (SAE); processamento de imagens; classificação de cobertura da terra; redução de dimensionalidade; dados hiperespectrais.

ABSTRACT

Deep learning opened new possibilities for hyperspectral data processing and analysis using multiple neural nets layers and can be used as a feature extraction tool. In this research, a pixel-based hybrid model is developed that integrates Stacked-Autoencoders (SAE) and Convolutional Neural Network (CNN) for hyperspectral image classification. The core of the integrated model (SAE-1DCNN) is an autoencoder that is improved by using convolutional layers in the encoding and decoding steps. This allows improving data discrimination in unsupervised training and reducing the processing time because it allows a feature-based description of the pixel's hyperspectral signature and takes advantage of the effectiveness of deep architecture based on the convolutional and pooling layers. As one-dimensional filters are applied, the processing time is considerably lower than when using 2D-CNN filters. In a first step, the SAE-1DCNN model is used for feature extraction and then these results are used in a final supervised classification step. Thus, in the first stage, the parameters of the net are adjusted using training samples and then, in the second stage, a fine-tuning approach followed by logistic regression based on the softmax activation function was applied for classification. Three aspects are analyzed in detail: the capacity of the model to exclude noisy bands, its ability to dimensionality reduction, and its potential to perform land cover classification based on hyperspectral data. Experiments were performed using different hyperspectral data sets: Indian Pines University of Pavia and Salinas, widely used by the scientific community, and a hyperspectral image captured at the Canguiri Farm of the Federal University of Paraná (UFPR) in Paraná-Brazil. To validate the proposed methodology, the obtained results were compared to traditional machine learning methods to verify the potential of the integration of autoencoders (AE) and convolutional nets. These obtained results showed similarity with traditional methods in terms of hyperspectral classification accuracy, however, they demanded less time for processing, therefore, the proposed methodology (SAE-1DCNN) is considered promising, solid, and can be an alternative for hyperspectral data pre-processing and processing.

Key-words: deep learning; convolutional neural network (CNN); stacked-autoencoders (SAE); image processing; land cover classification; dimensionality reduction, hyperspectral data.

RESUMEN

Deep Learning o aprendizaje profundo abrió nuevos desafíos para el preprocesamiento, procesamiento y análisis de datos hiperespectrales usando varias capas de redes neuronales y puede ser usado como herramienta de extracción de atributos. En esta investigación, se desarrolla un modelo híbrido basado en pixeles que integra *Stacked-Autoencoders* (SAE) y redes Neuronales Convolucionales (CNN) para clasificar datos hiperespectrales. Este enfoque uso un modelo basado en pixeles que integra *Convolutional Neural Networks* (CNN) y *Stacked-Autoencoders* (SAE). El núcleo del modelo integrado (SAE-1DCNN) es un Autoencoder (AE) mejorado que usa capas convolucionales en las etapas de codificación y decodificación. Esto permite mejorar la discriminación de datos a través de un entrenamiento supervisado y además reducir el tiempo en el procesamiento, pues permite una descripción de los atributos basad en la respuesta hiperespectral del pixel y aprovecha la efectividad de la arquitectura profunda en las capas convolucionales (convolutional) y de agrupamiento (pooling). En este modelo integrado se aplican filtros unidimensionales lo que permite que el tiempo en el procesamiento sea menor si se compara con los filtros bidimensionales 2D-CNN. En una primera etapa, el modelo SAE-1DCNN es usado para la extracción de atributos y en seguida, esos resultados son usados para la etapa final basada en la clasificación supervisada. De esta forma, en la primera etapa los parámetros de la red son ajustados usando las muestras de entrenamiento y después en la segunda etapa el enfoque conocido como *fine-tuning* fue aplicado para la clasificación de cobertura terrestre basado en regresión logística y la función de activación softmax. Tres aspectos son analizados en esta investigación, la capacidad del modelo para excluir bandas ruidosas, la capacidad para seleccionar las bandas redundantes y así reducir la dimensionalidad y el potencial para realizar la clasificación de la cobertura terrestre usando datos hiperespectrales. Los experimentos fueron realizados con diferentes conjuntos de datos hiperespectrales: Indian Pines, Universidad de Pavia y Salinas, ampliamente usados en trabajos científicos, y una imagen hiperespectral capturada en la Hacienda Canguiri de la Universidad Federal de Paraná (UFPR) en Paraná-Brasil. Para validar la metodología propuesta, los resultados obtenidos se compararon con métodos tradicionales de aprendizaje de máquina (*machine learning*) para verificar el potencial de la integración de *Autoencoders* (AE) y redes convolucionales. Los resultados obtenidos mostraron similitud con los métodos tradicionales en cuanto a la precisión de clasificación hiperespectral, sin embargo, exigieron menos tiempo de procesamiento, por lo que, la metodología propuesta (SAE-1DCNN) se considera prometedora, sólida y puede ser una alternativa para el pré-procesamiento y procesamiento de datos hiperespectrales.

Palabras-llave: *deep learning; convolutional neural network* (CNN); *stacked-autoencoders* (SAE); procesamiento digital de imágenes; clasificación de cobertura terrestre; reducción de dimensionalidad; datos hiperespectrales.

LIST OF FIGURES

LIST OF FRAMES

LIST OF TABLES

LIST OF GRAPHICS

ABBREVIATIONS

AA	- Average Accuracy
AE	- Autoencoders
ANN	- Artificial Neural Networks
AVIRIS	- Airborne Visible / Infrared Imaging Spectrometer
CNN	- Convolutional Neural Networks
DBM	- Deep Boltzmann Machines
DBN	- Deep Belief Network
FCN	- Fully Connected Network
GAN	- Generative Adversarial Networks
GPU	- Graphics Processing Unit
KIT	- Karlsruhe Institute of Technology
LSTM	- Long Short-Term Memory
OA	- Overall Accuracy
PA	- Producer’s Accuracy
ReLU	- Rectified Linear Unit
RNN	- Recurrent Neural Networks
ROSIS	- Reflective Optics System Imaging Spectrometer
SAE	- Stacked-Autoencoders
SAR	- Synthetic aperture radar
SVM	- Support Vector Machine
Tanh	- Hyperbolic Tangent
UA	- User’s Accuracy

CONTENTS

1 INTRODUCTION

Remote sensing became a useful and efficient tool for practical applications related to identifying, mapping, and monitoring changes on the Earth's surface (Zhu et al., 2020; Paoletti et al., 2018; Hassan et al., 2017) such as land use and land cover classification, change detection, environment monitoring and management and precision agriculture (Paoletti et al., 2018; Huang et al., 2018; Mahesh et al., 2015; Henits et al., 2016; Kussul et al., 2017). Although remote sensing is widely used nowadays, it experiences the impact of sensors and classification methods that improve the accuracy of its products (Zhu and Woodcock, 2014).

Among these sensors are those that enable the acquisition and processing of hyperspectral images, composed of hundreds of contiguous narrow spectral bands that provide valuable information to accurately differentiate objects and materials (Chen et al., 2016; Li et al., 2015). Increasing the spectral image resolution can be considered, initially, a gain in terms of information contents. Nevertheless, describing the spectral response of targets with several bands also increases redundancy in the data set and the necessity to use more sophisticated methods to process such data.

On the other hand, advances were also achieved in terms of the processing capacity of computers, which are useful in digital image processing and analysis. Better computers enabled using classification methods that were not possible before because of their need of high computing requirements. New methods were also proposed, including those based on Artificial Intelligence. For example, traditional Artificial Neural Networks (ANN) approach developed towards new machine learning methods, like Deep Learning or Hierarchical Learning, that allows extracting representative features from images. As shown in Chen et al. (2014); Geng et al. (2015); Wang et al. (2016), Deep Learning methods proved to be efficient in hyperspectral, multispectral and radar image processing. Combining such advances, and according to Zhu et al., (2017), improvements of the classification’s accuracy and the analysis of more complex data are expected especially exploring the ability of deep learning methods to extract attributes from hyperspectral images.

Several Deep Learning models can be applied for hyperspectral remote sensing image processing and classification, such as Convolutional Neural Networks (CNN) and Autoencoders (AE). Experiences reported in the literature on the use of Convolutional Neural Networks to classify hyperspectral data (Li et al., 2018; Paoletti

et al., 2018; Zhao and Du, 2016a) show robust and effective results. Two facts contribute to the good performance of Convolutional Neural Networks: Its deep structure (the model uses multiple layers to transform the input data into a suitable representation) and the convolution and pooling processes (Du et al., 2016). Other suitable Deep Learning models for satellite image processing are Autoencoders (AE), or Stacked-Autoencoders (SAE). This model is a symmetrical neural network that uses unlabeled inputs to build up a compressed feature representation from a high dimensional feature space (Cheng et al., 2017). Initially, autoencoders were considered a valuable alternative for dimensionality reduction but they can also be used to perform unsupervised classification because of its potential to compute more discriminative features from the original data (Paoletti et al., 2019; Wang et al., 2016; Zabalza et al., 2016). Furthermore, a trained autoencoder can be adapted for supervised classification, adjusting its parameters with the help of labeled training samples and techniques like supervised fine-tuning, that allows reducing classification errors using logistic regression (Lin et al., 2018).

Several researchers developed methods for land cover classification and dimensionality reduction based on CNN, SAE or Convolutional-AEs (Chen et al., 2014; Wang et al., 2017; Othman et al., 2016; Li et al., 2016; Zhao and Du, 2016b; Mei et al., 2019). However, most studies use 2D and 3D models and few use pixel-based models (1D), as reported by Hu et al., 2015 or Li et al., 2016, although 1D models are simpler and faster, in terms of computational complexity and effort, and they do not require very deep architectures in the training step (Kiranyaz et al., 2020). This means that 1D structures are easier to train and do not need sophisticated and expensive processing units (e.g., Graphics Processing Unit-GPU) to obtain robust and accurate results.

This study aims at exploring the potential of a 1D deep learning model based on stacked autoencoders (SAE) that includes convolutional layers, in the encoding and decoding phases, for pre-processing and classifying hyperspectral images. Three aspects are studied in detail: The potential use of the model to perform land cover classification, the exclusion of noisy bands, and dimensionality reduction. The expected advantages are increasing mapping accuracy and reducing the processing time.

1.1 RESEARCH AIMS

The aim of this research is to develop a pixel-based hybrid model that integrates Stacked-Autoencoders and Convolutional Neural Network for hyperspectral images classification. The specific aims are:

- To propose a method to include CNN models into the processing flux of SAE for hyperspectral images processing.
- To evaluate the proposed SAE-1DCNN method using different datasets, obtained with different hyperspectral sensors.
- To compare the performance of the SAE-1DCNN model to those achieved with other hyperspectral image classification methods.
- To verify if the SAE-1DCNN models allows identifying noisy bands within a hyperspectral image dataset.
- To verify if the SAE-1DCNN model allows reducing the dimensionality of hyperspectral images based on the feature selection capacity.

1.2 BACKGROUND

Deep learning rose as a promising alternative to process and specially classify images, including remote sensing images. The most well-known deep learning solutions were proposed to detect and identity objects in images, analyzing image regions (2D) and computing spatial features from such regions. The advantage of these deep learning approaches is that they minimize human interaction, providing automated solutions to the classification problem.

Traditional deep learning methods compute features based on the variation of the digital numbers within a given image region. Such features are used to describe the image and are expected to substitute the features that a human analyst would propose based on his knowledge. The experience proved that such alternative is feasible and that it is possible to describe the desired object with machine derived image features and use them to detect the object in other images. Among some works that use deep learning based on the variation of digital numbers to detect objects are: Li et al., 2017; Hu et al., 2015.

The success of deep learning applied to image analysis encouraged its use in hyperspectral image classification. As the information of a pixel is stored in a high dimensional vector, it is possible to analyze variations along the spectrum in detail to deduce the most probable class within a classification. For this purpose, instead of deriving spatial features, it is possible to compute spectral features from the pixel's hyperspectral vector using a 1D approach. The use of 1D deep learning models in image processing, although simpler, is not common, because most studies are devoted to RGB images with a reduced set of spectral bands. 1D model may not be appropriate to process and classify an RGB image, but become interesting when dealing with hyperspectral data, where a pixel contains digital values related to many narrow and contiguous spectral bands. The use of 1D models discards the inclusion of context but can be useful to analyze the spectral signature of each pixel.

A common problem when dealing with hyperspectral data and statistical classification methods is related to the number of training samples. The number of training samples needs to be increased, as the number of spectral bands grows. Nevertheless, it is difficult (sometimes impossible) to increase the number of samples in real situations. Therefore, dimensionality reduction procedures are applied before the classification step to reduce the number of input bands and reduce redundancy in the data set.

Deep learning methods, such as convolutional networks, are not based on the assumption of statistical distributions or parameters that need to be estimated with training samples, which is an advantage. On the other hand, they also need large number of samples to compute reasonable features and feature selection would be recommended in a pre-processing step. This problem can be solved applying autoencoders, which are well-known methods for feature selection. So, the combined use of autoencoders and convolutional networks seems to be a reasonable solution for the pre-processing (identification of noisy bands and feature selection of representative bands for dimensionality reduction) and processing (land cover classification) of hyperspectral data. It is expected that the combined model will be efficient to describe the spectral signature of the classes and enable a more accurate classification of the pixels, by combining the feature computation capacity of convolutional networks with the dimension reduction capacity of autoencoders.

1.3 ORGANIZATION OF THE DOCUMENT

This document describes a hybrid approach integrating the SAE and CNN models for pre-processing and processing of hyperspectral datasets and is organized as follows: The first topic consists of the introduction, research objectives and background. In the second chapter it is presented a review of the literature on the use of deep learning methods for remote sensing image processing. Special attention is paid to the fundamental concepts regarding artificial neural networks and their architecture, as well as the use of unsupervised, supervised, and hybrid learning in remote sensing. It is also discussed the need of the exclusion of noisy bands, dimensionality reduction, as well as the land cover classification and performance assessment of the results. In the third chapter the proposed methodology is described in detail. The proposed autoencoder enhanced with the 1D convolutional layers is described first and then the experiments aimed at verifying the proposed solutions are presented. Finally, the results are discussed in the fourth chapter, followed by the conclusions and recommendations in the fifth chapter.

2 LITERATURE REVIEW

In this work, deep learning tools, such as convolutional neural networks and autoencoders, are used to classify hyperspectral images. Therefore, it is first introduced the principles of neural nets and how this concept is extended to build up deep learning models: the basics of the artificial neuron, its combination into complex neural nets and the principles of autoencoders and convolutional nets. Finally, a survey on the application of such deep learning tools is discussed.

2.1 ARTIFICIAL NEURAL NETWORK

The basic idea of Artificial Neural Networks (ANN) was proposed by McCulloch and Pitts in 1943 to understand and model the structure and functioning of biological neural networks. They perceived that information flows through a neural net and that the *stimuli* are transmitted and processed within a net of multiple elements, called neurons (Negnevitsky, 2005). The function of neurons in the human brain is to process information and pass it to other neurons through the synapses. So, each neuron receives inputs from other neurons and produces an output.

A neuron can be mathematically modeled according to Equations 1 and 2:

$$u_k = \sum_{j=1}^{n} w_{k,j} \cdot x_j \tag{1}$$

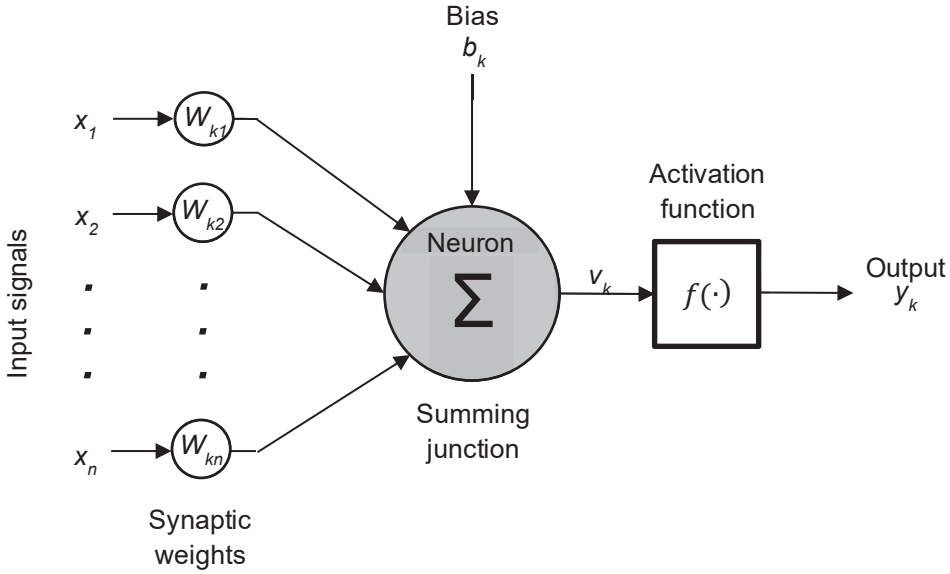
$$y_k = f(u_k + b_k) \tag{2}$$

where $x_1, x_2, \ldots, x_n$ represent input signals, which are combined with the synaptic weights $w_1, w_2 \ \ldots w_n$; each neuron produces an output y_k as a function of the inputs, that is modulated by an activation function $f(\cdot)$ and can be passed on to other neurons; according the expression 2.

In the mathematical model of a neuron, as displayed in Figure 1, the inputs (x) are added, applying weights (w), to compute an output. It is included an external *bias* to the summation, identified as b_k. The *bias* b_k can increase or decrease the result of the weighted sum that is passed to the activation function. The result of the

weighted sum of the inputs is then evaluated using an activation function, that transforms this result into a modulated output (y) (Atkinson and Tatnall, 1997).

FIGURE 1 - DIAGRAM OF A NEURON



FONT: Adapted from Haykin (2008).

According to McCulloch and Pitts (1943), in the simple neuron model, the neuron computes the weighted sum of the input signals and compares the result with a threshold value, θ. If the net input is less than the threshold, the neuron output is -1. But if it is greater than or equal to the threshold, the neuron is activated, and its output is +1. In a more general approach, the output can vary between 0 and 1 (Haykin, 2008).

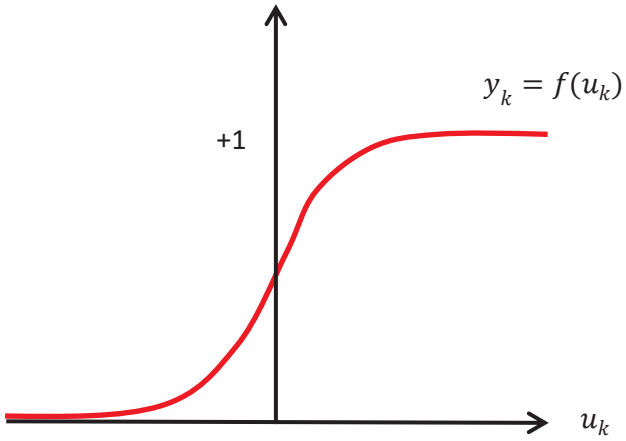
The output of a neuron is controlled by the activation function that is applied to make the network dynamic and add the ability to extract complex information from the input data and represent functional mappings for the output data (Sharma et al., 2020). The selection of the activation function depends on the domain to be solved such as: object recognition and classification, segmentation, speech recognition, cancer detection systems, fingerprint detection, weather forecast, machine translation, among others. Based on this domain, there are several types of activation function such as, Linear, Sigmoid, Hyperbolic Tangent (Tanh), Rectified Linear Unit (ReLU), Softmax, among others.

The sigmoid activation function (Figure 2) is the most used because it exhibits a balance between linear and non-linear behavior (Haykin, 2008). This function is defined by

$$y_k = \frac{1}{1 + e^{-u_k}} \tag{3}$$

where, u_k represent input signals; y_k are the outputs obtained after training the neural network.

FIGURE 2 - REPRESENTATION OF SIGMOID ACTIVATION FUNCTION



FONT: Adapted from Nwankpa et al., 2018

The Softmax function (Equation 4) has similar characteristics as the sigmoid function and is used to compute the probability distribution from a vector of real numbers. The Softmax function produces an output that ranges between 0 and 1, with the sum of the probabilities equal to 1 (Nwankpa et al., 2018).

$$y_i = \frac{e^{u_i}}{\sum_{j=1}^{n} e^{u_j}} \tag{4}$$

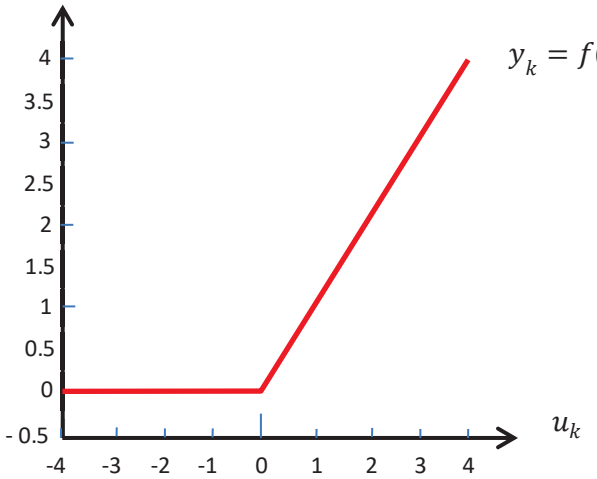
where, u_i , u_j represent input signals; y_i are the outputs obtained after training the neural network

The Sigmoid and Softmax functions are widely used and successfully applied for classification tasks in logistic regression models. However, the difference between these two functions is that the Sigmoid function is used for binary classification, while Softmax is used for multivariate classification tasks (Sharma et al., 2020).

A non-linear activation function that is used only in hidden layers and not in external layers of neural networks is the ReLU activation function (Nwankpa et al., 2018). The ReLU function (Figure 3) can be used for object classification and pattern recognition when combined with other activation functions (Sigmoid or Softmax) in the output layers of network (Wang et al., 2018). It reduces the vanishing gradient problem observed in other activation functions (Hara et al., 2015) by discarding negative inputs. This activation function applies a threshold to each input, according to equation 5.

$$y_k = \max(0, y) = \begin{cases} u_k \text{ if } u_k \geq 0 \\ 0 \text{ if } u_k < 0 \end{cases} \quad (5)$$

FIGURE 3 - REPRESENTATION OF ReLU ACTIVATION FUNCTION

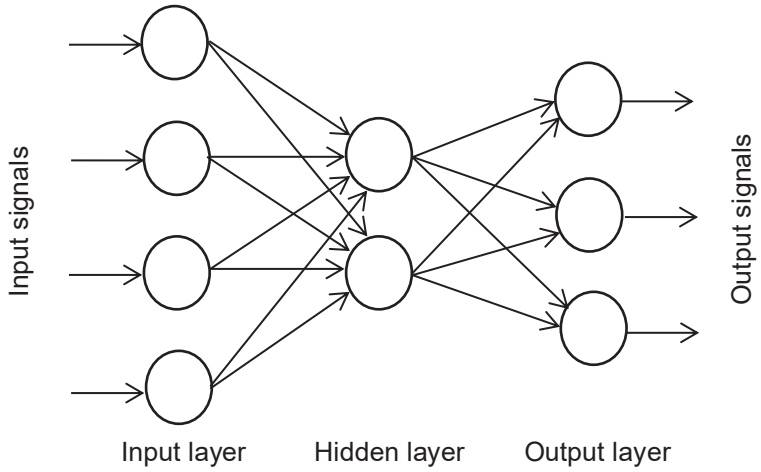


FONT: Adapted from Nwankpa et al., 2018

Several artificial neurons can be arranged to build up an artificial neural net, where the inputs of a neuron can produce an output that is passed to other neurons, which uses it as a new input. Although the natural neural net may be more complex, an artificial neural network is organized in layers of neurons. A typical architecture of a simple ANN is displayed in Figure 4. According to this figure, the ANN is composed of a hierarchy of layers (input, hidden and output layers) where the neurons are

organized. The input layer just reads the input data and pass the signal to the series of hidden layers. The outputs of a hidden layer can be used as inputs by the next one and the process can be repeated for several layers, passing information along the network (Haykin, 2008).

FIGURE 4 - ARCHITECTURE OF AN ARTIFICIAL NEURAL NETWORK



FONT: Adapted from Negnevitsky (2005).

The necessary number of layers and neurons, the architecture of the network, for a given problem is still an open question. The more neurons and layers are used, the network can solve more complex problems. Nevertheless, overfitting may happen if the architecture is too complex for the problem.

Neural nets are widely used in remote sensing image processing, specially to perform classification. Examples can be found in Pal (2009), Ul Haq et al., 2011, Gong et al. (2011), Garcia-Salgado et al., 2018, Panda et al. (2010), Neagoe et al. (2012), Ahmed et al. (2013), Isik et al. (2013), or Jia et al. (2015). A more comprehensive description of such examples is listed in Frame 1.

FRAME 1 - WORKS PERFORMED USING ANN APPLICATIONS

Area	Reference	Summary
Land cover classification	Pal, 2009	A supervised classifier (ELM) based on a single hidden layer neural network for land cover classification for multispectral and hyperspectral images. In this work, a comparison is made with a typical neural network based on the backpropagation algorithm. The results of this work suggest that ELM provides greater classification accuracy comparable to a backpropagation neural network for both data sets and beyond, it takes less time to process.
	Ul Haq et al., 2011	An approach for hyperspectral data classification using adaboosting of artificial neural networks based weak classifiers was proposed using hyperspectral images. The adaboost algorithm employs an iterative approach which combines weak classifiers to approximate a Bayes classifier. In the approach, several neural networks based weak classifiers were used to make one strong classifier and each weak classifier contains only one hidden layer. As the weak classifiers are simple, therefore, these does not require a lot of time for training and, therefore, are time efficient.
	Gong et al., 2011	A classification model based on an optimized artificial immune network (OPTINC) was developed to classify land use and land cover. The OPTINC model was evaluated using high spatial resolution QuickBird data and LiDAR data for residential areas, a HyMap hyperspectral image for suburban areas. A decision tree, a multi-layer feed-forward neural network with a backpropagation algorithm, and aiNet were also tested for comparison. The OPTINC model outperformed the other models with higher accuracy and more spatially cohesive land cover classes with limited salt-and-pepper noise.
	Garcia-Salgado et al., 2018	A classification system that can be used for both multispectral and hyperspectral data was proposed. The designed classification system is composed of a novel parallel feature extraction algorithm, which utilizes a cluster of two graphics processing units in combination with a multicore central processing unit (CPU), and an artificial neural network (ANN) particularly devised for the classification of the features ensued by the implemented feature extraction method.

FRAME 1 - WORKS PERFORMED USING ANN APPLICATIONS

(continuation)

Monitoring and change detection	Panda et al., 2010	The strength of key spectral vegetation indices for agricultural crop yield prediction using neural network techniques was investigated. The vegetation indices that were analyzed the corn crop yield were: normalized difference vegetation index (NDVI), green vegetation index (GVI), soil-adjusted vegetation index (SAVI) and perpendicular vegetation index (PVI). These four indices were investigated for 3 years (1998, 1999 and 2001) and for the pooled data of these 3 years and were tested with aerial images. The PVI-based models provided average accuracy of corn yield prediction than other forecast models in 1998, 1999 and 2001. In this context, this work verified the usefulness of the application of ANNs as a tool for predicting agricultural yields with high accuracy.
	Neagoe et al., 2012	An approach for land cover change detection in remote sensing imagery was presented. This approach supervised neural network change detection techniques versus statistical supervised were considered and evaluated. Firstly, supervised neural classifiers were evaluated: Multi-layer Perceptron (MLP), Radial Basis Function Neural Network (RBF) and Self Organizing Map (SOM), while the statistic classifiers applied were: Bayes and Nearest Neighbor. Secondly, unsupervised change detection techniques: Self Organizing Map (SOM) (neural grouping), versus K-means (statistic grouping) and Fuzzy C-means (fuzzy grouping) were investigated. The proposed model of change detection in multispectral satellite images has two main processing stages: (a) feature selection, using three techniques: concatenation of corresponding pixels (CON), computation of absolute differences between corresponding pixels (ADIP), and computation of absolute differences between reflectance ratios of corresponding pixels (ADIRR) and (b) classification, using the above-mentioned supervised or non-supervised models for obtaining two classes: “change” and “no change”. The better result using supervised techniques was CON-MLP, while that the unsupervised techniques was ADIP-SOM. The results prove the advantage of the neural network change detection techniques over the statistical and fuzzy.

FRAME 1 - WORKS PERFORMED USING ANN APPLICATIONS

(continuation)

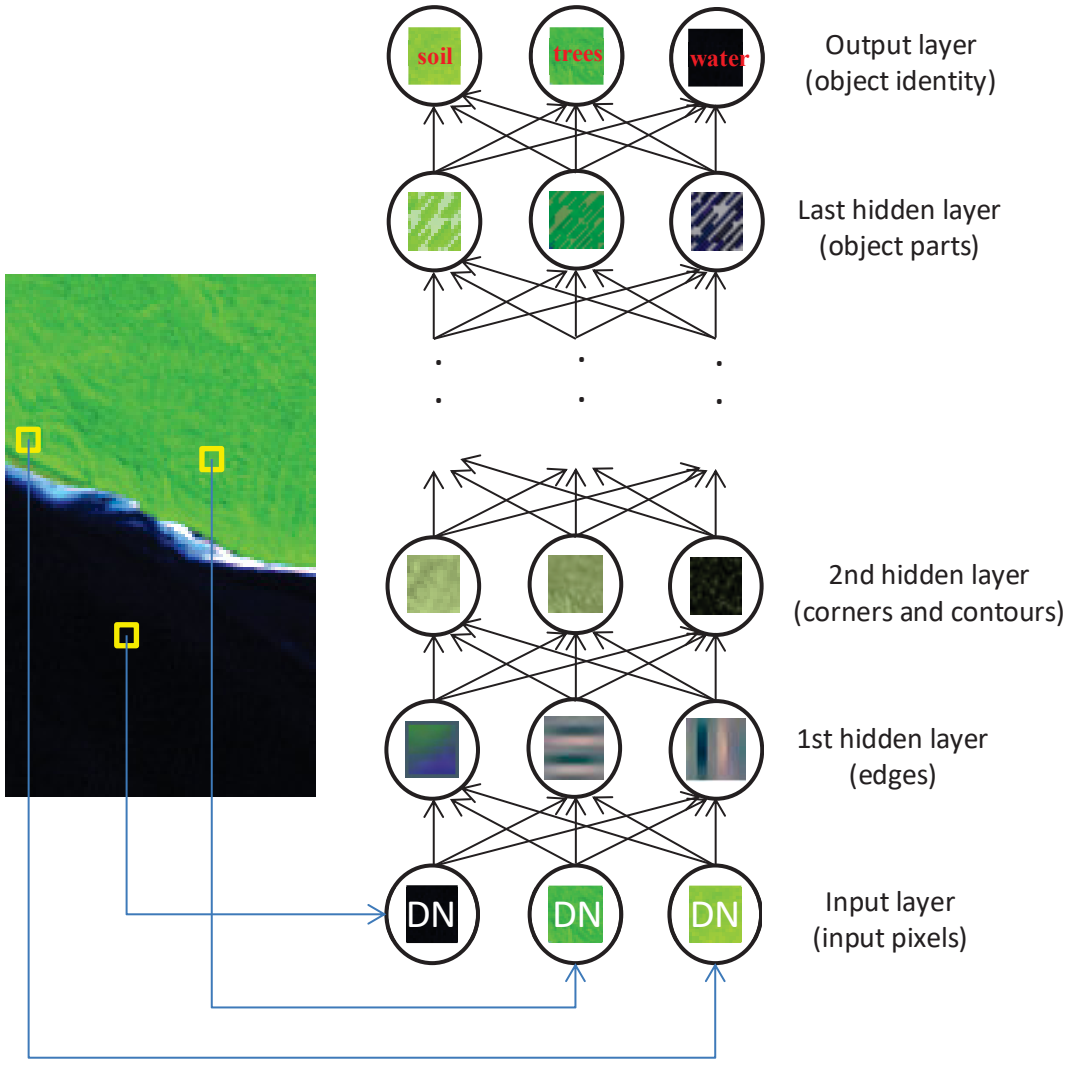
Monitoring and change detection	Ahmed et al., 2013	The dynamics of land cover and its impacts on land surface temperature in fast-growing megacities in developing countries were analyzed using Landsat satellite images from 1989, 1999 and 2009. This study was based on three objectives: (1) identified patterns of land cover changes between periods and investigated their impacts on land surface temperature; (2) applied an ANN to simulate land cover changes for 2019 and 2029; (3) estimated their impacts on land surface temperature in respective periods. The results would help to quantify the impacts of different scenarios (for example, vegetation loss to accommodate urban growth) on land surface temperature and to design appropriate policy measures.
	Isik et al., 2013	A hybrid model based on Artificial Neural Networks (ANNs) and Soil Conservation Service (SCS) Curve Number (CN) was developed to predict the effect of changes in land use/cover (LULC) on daily streamflows. The model used data from LULC, hydrologic soil groups and climatic factors, such as temperature and precipitation, in order to replicate the hydrologic response of a watershed, in addition, incorporated data from neighboring watersheds for training, validation and testing purposes. For the development of this study, the authors used aerial images with spatial resolution of 1m. The results indicated that the hybrid model developed was able to predict the increases in average flow and flashiness for the urban and pasture dominant scenarios and for the forest dominated scenarios, the model predicted more stable hydrology (less flashy) with lower average flows.
	Jia et al., 2015	A reliable estimation algorithm to operationally produce a high-quality global Fractional vegetation cover (FVC) product from the Moderate Resolution Imaging Spectroradiometer (MODIS) surface reflectance was developed. For the development of this algorithm, Landsat TM / ETM+ and MODIS images were processed. The results obtained in this study indicated that the spatial and temporal continuity of the estimates from the proposed method was superior to that of the other product

FONT: The author (2021).

2.2 DEEP LEARNING

Deep Learning is a hierarchical technique that allows analyzing and processing data based on various artificial neural networks. This model emerged as a learning tool to solve a wide range of tasks related to computer vision, visual object recognition, natural language processing, logical reasoning, handwriting recognition, audio processing, information retrieval, robotics, among others (Markoff, 2012; Ball et al., 2017). However, this technique has still several unique challenges related to remote sensing (hyperspectral, multispectral and radar) and its applications, such as land cover classification, change detection, data fusion and object recognition (Kussul et al., 2017; Ball et al., 2017; Lin et al., 2018).

FIGURE 5 - ILLUSTRATION OF THE DEEP LEARNING MODEL



FONT: Adapted from Goodfellow et al., 2016.

Figure 5 displays an illustration of the use of deep learning to classify a satellite image. In this example, not only a pixel, but a region around a pixel is analyzed to compute spatial features. The input layer receives the digital numbers of the pixels within the region that are combined in the hidden layers, which allows identifying features like edges, corners, and contours, that are later used to recognize and map the objects. The result is then presented in the output layer of the deep learning model.

The novelty of such approach is that the analyst does not need to specify or propose the features that are used to solve the problem and how are they combined, but the system computes a series of features, sometimes more than necessary, selects the most significant ones and organizes their combination according to the proposed aim. In this sense, a high level of automation is introduced. As each layer is composed of artificial neural nets, it is said that the nets are stacked, building a deep representation of the data.

2.2.1 Deep Learning approaches

According to Deng (2014), there are three approaches based on Deep Learning technique: unsupervised or generative, supervised and hybrid learning (supervised and unsupervised).

2.2.1.1 Unsupervised or generative learning

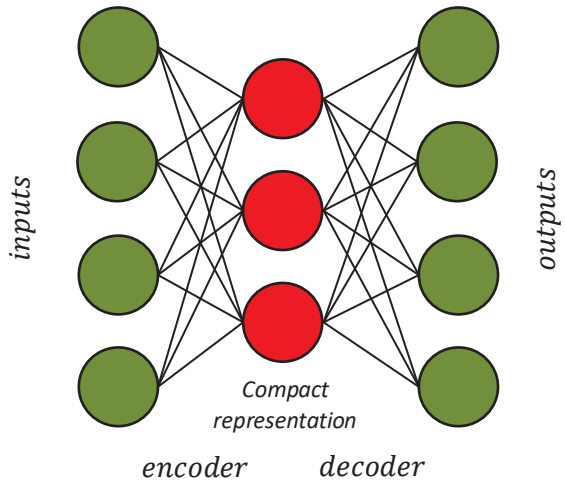
This approach refers to learning methods in which the output classes is not “a priori” known and the input data is used to detect common patterns within the data set. Therefore, it is not necessary to provide labelled samples to guide the solution. When used in generative mode, this approach can also be intended to characterize the joint statistical distributions of visible data and their associated classes (Deng, 2014).

The most common models for unsupervised deep learning include Autoencoders (AE) or Stacked-Autoencoders (SAE), Deep Boltzmann Machines (DBM) and Deep Belief Network (DBN). As stacked autoencoders are used in the present study, they are described below in detail.

2.2.1.1.1 Stacked-Autoencoders (SAE)

To understand a Stacked Autoencoder (SAE) it is necessary to understand how single Autoencoders (AE) work. The autoencoder model has a symmetric architecture and is designed to learn a representation of compressed and distributed data sets. AE are made up of two parts, as displayed in Figure 6: the first phase (encoder) compresses the input data to a relatively shorter representation, while in the second phase (decoder) the compressed data are used to reconstruct the original input. So, the AE computes a compact representation preserving the information that still allows the reconstruction of the original data, minimizing information loss (Khan et al., 2019).

FIGURE 6 - AUTOENCODER NETWORK ILLUSTRATION



FONT: The author (2021).

During the encoder stage, an input vector $x \in R^N$ is mapped to the hidden representation or code, h, through a non-linear activation function f. If the network has a simple hidden layer, then h will be expressed by equation 6.

$$h = f(Wx + b) \tag{6}$$

where, W is the weight matrix to be estimated at the training stage (learned) and b is a bias vector.

The decoder stage maps code h (the hidden representation) to the output x'. This stage performs a reconstruction or approximation of the input vector through a non-linear activation function, like the previous stage, by means of equation 7:

$$x' = f(W'h + b') \tag{7}$$

where, W' and b' are the weight matrix and bias vector of the reconstruction layer with a non-linear activation function f.

Normally, the encoding and decoding weight matrices are linked, so that $\boldsymbol{W'}$= W^T (T is the operator of the transposed matrix), this means that, the same weight matrices are used for encoding the input and decoding the latent representation (Zhu et al., 2017; Ball et al., 2017).

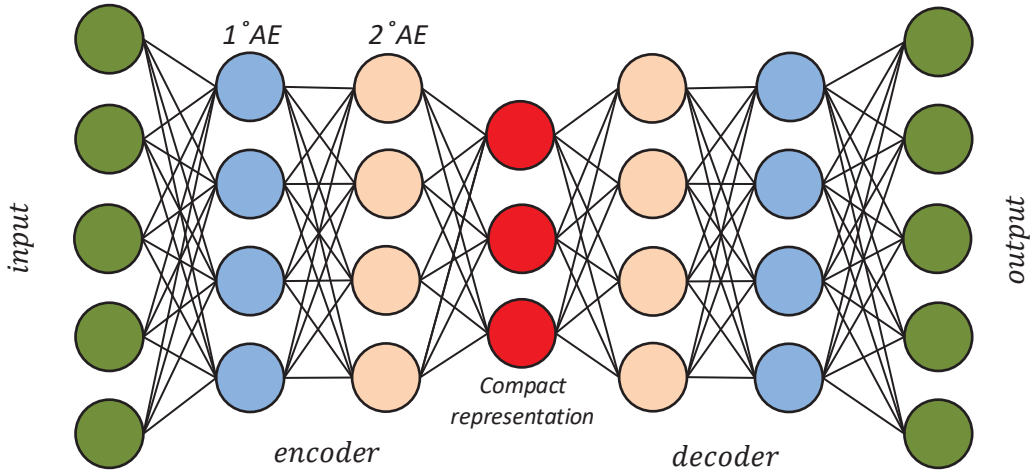
The compact representation obtained (latent representation) with the AE network is often understood as a feature vector that can be used for several purposes, including clustering, indexing and dimensionality reduction (Khan et al., 2019)

To determine the optimized parameters of an AE model, the difference between the reconstructed outputs (output layer) and the inputs (input layer) must be minimized (Zabalza et al., 2016), according to equation 8:

$$\underset{W_x, W'_x, b, b'}{argmin}[error(x, x')] \tag{8}$$

Autoencoders can be combined (stacked) to build up a more complex network: A Stacked-Autoencoders network (SAE). In the SAE model the outputs of one Autoencoder is used to feed the next one. An example of an SAE architecture is shown in Figure 7. In this figure, the SAE model is made up of three hidden layers for the encoder stage and another three hidden layers for the reconstruction or decoder stage.

FIGURE 7 - SAE ARCHITECTURE



FONT: The author (2021).

In the encoding phase, each following layer is smaller, which allows reducing processing time and a deep representation of the spectral data based on features computed from the inputs (Ma et al., 2019). A great advantage of the SAE model is that it allows unsupervised training, which avoids the necessity of labelled data.

According to Zabalza et al. (2016) there is an iterative update of the coefficients (w, b) of each internal AE in the training stage of an SAE model. This update progressively reduces the error between the input pixel and the reconstructed pixel (output) until reaching a threshold based on the criteria of equation 8. This means that, in an SAE model, the pre-trained parameters of an internal AE layer are frozen for the training of the subsequent layers (Yue et al., 2016).

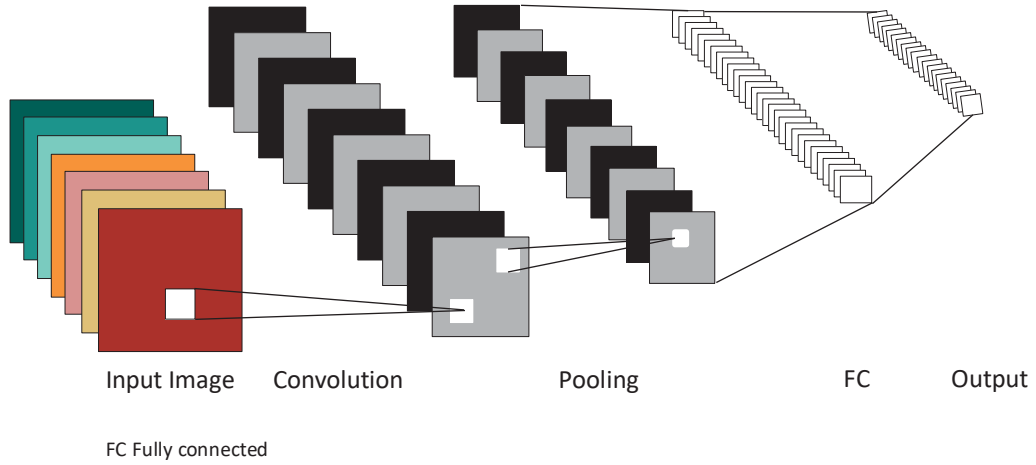
2.2.1.2 Supervised learning

Supervised learning is based on prior knowledge of the expected classification of the input data. In other words, to train the system it is necessary to present the expected result for each input. The system adjusts the weights aiming at reproducing the expected output based on the given inputs (Khan et al., 2019). According to Schmidhuber (2015), the aim of supervised learning is to find the best weights minimizing the global error rate. Supervised training can be used to solve classification and regression problems. The supervised deep networks are also known as discriminative networks, and the most common architectures for this type of learning are: CNN and Recurrent Neural Networks (RNN).

2.2.1.2.1 Convolutional Neural Networks (CNN)

Convolutional Neural Nets (CNN) are the most representative and widely used supervised neural networks model to solve image analysis problems. CNN is a deep learning model designed to process images using the regular organization of the pixels (Ma et al., 2019 and LeCun et al., 2015). It is composed of three different layers: the convolutional layer, the pooling layer, and the full connected layer (Figure 8). The Convolutional layers can be understood as 2D filter banks that transform an input image into a series of new images, highlighting specific patterns. As each filter produces a new image, the data volume increases. Therefore, a pooling layer is used to reduce the size of the output through non-linear operations. The full connected layer takes the output of the pooling layer as input to compute the final output (LeCun et al., 2015).

FIGURE 8 - CNN ARCHITECTURE



FONT: Adapted from Kumar et al. (2020)

According to Ma et al. (2019), in a CNN model, the input vector is convolved with a set of K kernels with the weight matrices $W = \{W_1, W_2, \ldots, W_K\}$ and biases are added $\gamma = \{b_1, b_2, \ldots, b_K\}$, each generating a new feature map X_k, according to equation 6. The output of each convolution is modulated applying a non-linear transform $f(\cdot)$, and the same process is repeated for every convolutional layer (Equation 9).

$$l: X_k^l = f(W_k^{l-1} * X^{l-1} + b_k^{l-1}) \quad (9)$$

Various applications of CNN-based hierarchical learning with robust and effective results prove that it is useful for automatically discovering relevant contextual features in image categorization problems (Maggiori et al., 2016). Having as most important applications: object detection and land cover classification on multispectral, hyperspectral and radar images as shown in Chen et al. (2014); Geng et al. (2015); Wang et al. (2016); Paoletti et al., (2018).

2.2.1.3 Hybrid learning

According to Deng (2014), hybrid learning uses the results obtained from a previous unsupervised network as basis to produce a more efficient network through the optimization or/and regularization of the parameters learned in a previous training. The improvement refers to the fact that generative models can provide excellent initialization points for highly nonlinear parameter estimation problems; and regularization refers to the generative models that can effectively control the complexity of the general model (Deng, 2014). For example, unlabeled data can be introduced into an unsupervised network that can learn the more significant features and clusters. This knowledge can later be used to train a classifier with labeled samples by supervised learning.

There are several examples of the use of hybrid architectures in the field of remote sensing in the scientific community. Some of these works are summarized in Frame 2.

FRAME 2 - RESEARCHS PERFORMED USING THE DEEP LEARNING HYBRID APPROACH

Area	Reference	Summary
Classification	Zhang et al. (2018)	A hybrid classification system is proposed that combines CNN (a context-based classifier with deep architectures) and MLP (a pixel-based classifier with surface structures) using a rules-based decision fusion strategy. This method was tested on aerial images with 50 cm spatial resolution and four multispectral bands (red, green, blue and near infrared) from both an urban scene and a rural area.

FRAME 2 - RESEARCHS PERFORMED USING THE DEEP LEARNING HYBRID APPROACH
(continuation)

Classification	Wu et al. (2016)	A new hybrid architecture was developed combining the AE network and the Fisher vector (based on the pooling layer) to automatically learn the representative and discriminative characteristics in a hierarchical way for the land cover classification. This hybrid model was tested on two images. The first image is an area (UC Merced) with a spatial resolution of 30 cm and the second is a satellite image collected from Google Earth that are sampled at 4 different scales.
	Ma et al. (2015)	A feature learning algorithm (Contextual Deep Learning) for classifying hyperspectral images is proposed. This algorithm can extract spectral and spatial features based on the Stacked Autoencoders (SAE) model. SAE is a hybrid learning method, which includes supervised and unsupervised steps. To test this algorithm, hyperspectral images were used: Indian Pines, Salinas and University of Paiva.
Object detection	Chen et al. (2014)	A hybrid model based on the Deep Learning network for vehicle detection in satellite images is presented. This model is divided into two parts. The first refers to extracting the attributes hierarchically using the convolutional and max-pooling layers (CNN algorithm). And the second is an MLP classifier, which classifies the extracted attribute data. The satellite images used for testing this algorithm were collected from Google Earth.
	Pashaei et al. (2019)	A hybrid system for the classification of accident images is proposed. This system uses a CNN for feature extraction and a mixture of extreme learning machines (ELM) for classification. The system (CNN-ELM) is developed through two tasks. For the first task, the outputs of the last max-pooling layer of a Convolution Neural Network (CNN) are used to extract the hidden features automatically. For the second task, a mixture of advanced variations of Extreme Learning Machine (ELM) including basic ELM, constraint ELM (CELM), On-Line Sequential ELM (OSELM) and Kernel ELM (KELM), is developed.

FONT: The author (2021).

2.2.2 Deep Learning in Remote Sensing

Interest on deep learning algorithms is increasing exponentially in the scientific community, mainly in the field of remote sensing and image processing. According to Ma et al. (2019), CNNs are established as the most widely used deep learning models and proved their ability to compute spectral-spatial features of images to solve image analysis challenges, like object detection, land cover and land use classification, segmentation, change detection, time series analysis, among others. However, there are other deep learning models that can be applied for the analysis and processing of remote sensing data, among which are: AE, SAE, RNN, Long Short-Term Memory (LSTM), deep belief networks (DBN), Fully connected network (FCN), generative adversarial networks (GAN), restricted Boltzmann machine (RBM), etc.

Figure 9 shows a taxonomy related to deep learning in remote sensing according to Ma et al., (2019), allowing the identification of different topics ranging from pre-processing to accuracy assessment in data analysis. In addition, Frame 3 presents a summary of the deep learning applications in remote sensing. For example, Xu et al., (2020), developed a general deep learning solution based on Convolutional Neural Network (CNN) and Stacked-Autoencoders (SAE) for atmospheric correction and target detection using multiple hyperspectral scenes. Zabalza et al., 2016, presented a deep model based on Stacked-Autoencoders for feature extraction in hyperspectral remote sensing (dimensionality reduction). Jing et al., 2020 proposed a novel deep learning architecture for change detection composed of a Trisiamese subnetwork and a Long Short-Term Memory (LSTM) subnetwork that fully utilizes the spatial, spectral and multiphase information and improves the change detection capabilities for very high-resolution multispectral imaging. Zhang et al., 2019, implemented a deep model based on a multilayer perceptron (MLP) and convolutional neural network (CNN) by via a Markov process involving iterative updating for land cover and land use classification using multispectral images. Kemker et al., 2018, implemented a Deep convolutional neural network for semantic segmentation using real and synthetic multispectral data. Paoletti et al., 2018, proposed a 3D-convolutional neural network (CNN) model that uses both spectral and spatial information for hyperspectral data classification. However, most of these works use deep models based on a single model and few

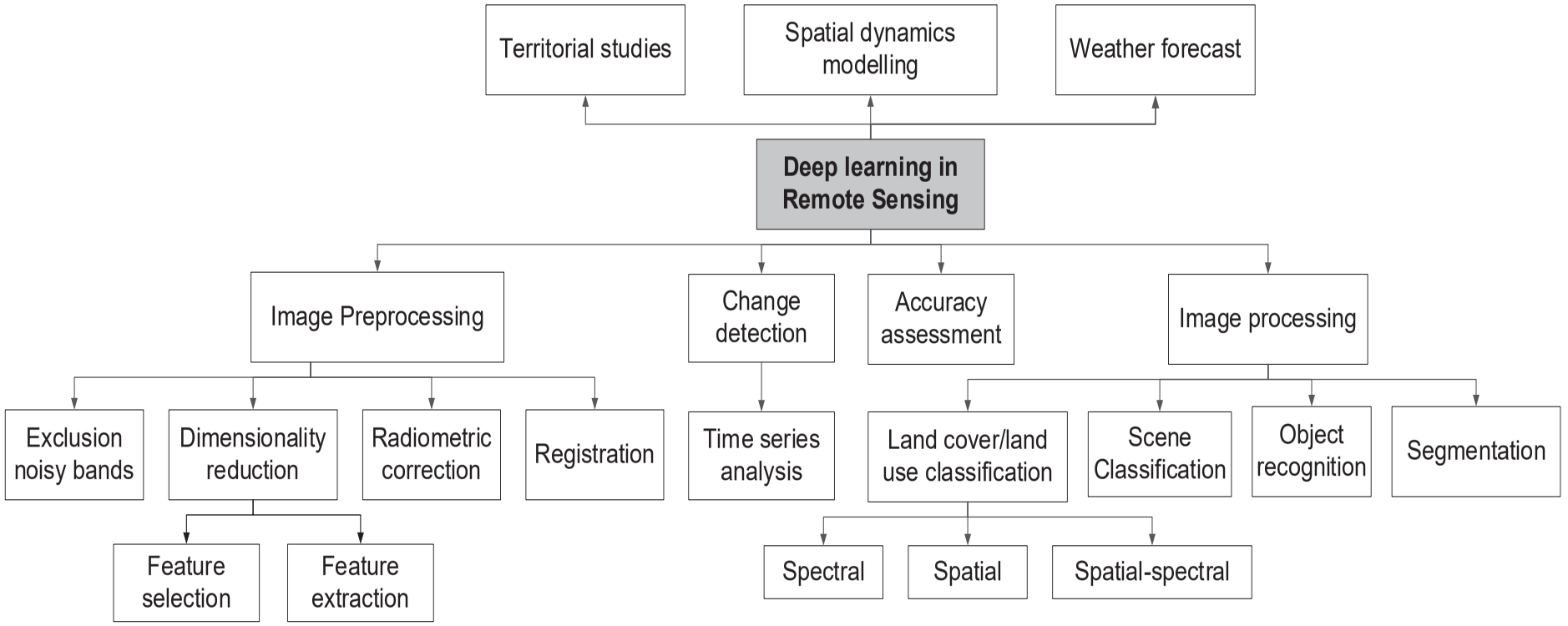
integrate several models for processing and all these works are used for a specific application such as: Radiometric Correction, Dimensionality Reduction, Change Detection, Land Use and Cover Classification, Semantic Segmentation, etc. The combination of several deep learning models for solving not only one problem in remote sensing applications is still new, but a promising field of research.

FRAME 3 - SUMMARY OF DEEP LEARNING TAXONOMY IN REMOTE SENSING

Area	Reference	Model
Image fusion	Xing et al., 2018	Stacked Sparse Autoencoders (SSAE)
Image registration	Merkle et al., 2018	Generative Adversarial Networks (GAN)
Radiometric correction	Xu et al., 2020	Convolutional Neural Network (CNN) and Stacked-Autoencoders (SAE)
Dimensionality reduction / feature extraction	Zabalza et al., 2016	Stacked-Autoencoders (SAE)
Change detection	Gong et al., 2017	Sparse Autoencoders and Convolutional Neural Network (CNN)
	Jing et al., 2020	Long Short-Term Memory network (LSTM)
Time series analysis	Wang et al., 2019	Bidirectional Long Short-Term Memory network (Bi-LSTM)
Land use and land cover classification	Zhang et al., 2019	Multilayer Perceptron (MLP) and Convolutional Neural Network (CNN)
Land cover classification	Sharma et al., 2017, Paoletti et al., 2018	Convolutional Neural Network (CNN)
	Chen et al., 2015	Deep Belief Network (DBN)
	Mei et al., 2019	Convolutional Autoencoder (CAE)
Land use classification	Feng et al., 2019	Convolutional Neural Network (CNN),
	Huang et al., 2018	Semi-Transfer Deep Convolutional Neural Network (STDCNN)
Accuracy assessment	Xing et al., 2018	Convolutional Neural Network (CNN)
Human detection	Kim and Moon, 2015	Deep Convolutional Neural Networks (DCNN)
Object recognition	Diao et al., 2015	Deep Belief Network (DBN)
	Sumbul et al., 2017	Convolutional Neural Network (CNN)
Semantic Segmentation	Basaeed et al., 2016	Convolutional Neural Network (CNN)
	Kemker et al., 2018	Deep Convolutional Neural Networks (DCNN)
Weather forecast	Hossain et al., 2015	Stacked Denoising Auto-Encoders (SDAE)
	Wimmers et al., 2019	Convolutional Neural Network (CNN)
Spatial dynamics	Reddy and Prasad, 2018	Long Short-Term Memory network (LSTM)
Territorial studies	Kussul et al., 2017	Convolutional Neural Network (CNN)

FONT: The author (2021)

FIGURE 9 - DEEP LEARNING TAXONOMY IN REMOTE SENSING



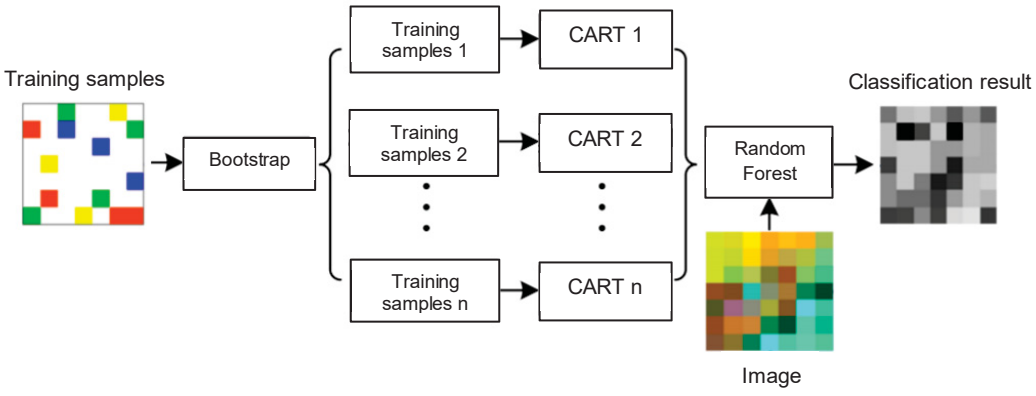
FONT: Adapted from Ma et al. (2019)

2.3 OTHER ALGORITHM FOR FEATURE EXTRACTION

2.3.1 Random Forest (RF)

The RF is an ensemble of many independent individual classification and regression tree (CART) to make a prediction (Breiman, 2001). In this sense, this algorithm is a combination of tree classifiers in which each classifier is obtained using a random vector sampled independently from the input vector, and each tree casts a unit vote for the most popular class to classify an input vector (Breiman, 2001). The creation of each decision tree that makes up the forest is the key to the success of RF and two steps are taken into consideration in the random selection process (Breiman, 2001). The first step uses a bootstrap strategy, where two thirds of the samples (in-bag samples) are used to train trees and the remaining one third (out-of-bag samples) is used for internal cross-validation to assess the classification accuracy. The second step of random sampling is to determine the split conditions for each node in the decision tree (Breiman, 2001). Figure 10 shows a schematic diagram of the RF algorithm for classification.

FIGURE 10 - SCHEMATIC DIAGRAM OF RANDOM FOREST FOR IMAGE CLASSIFICATION



FONT: From Feng et al. (2015)

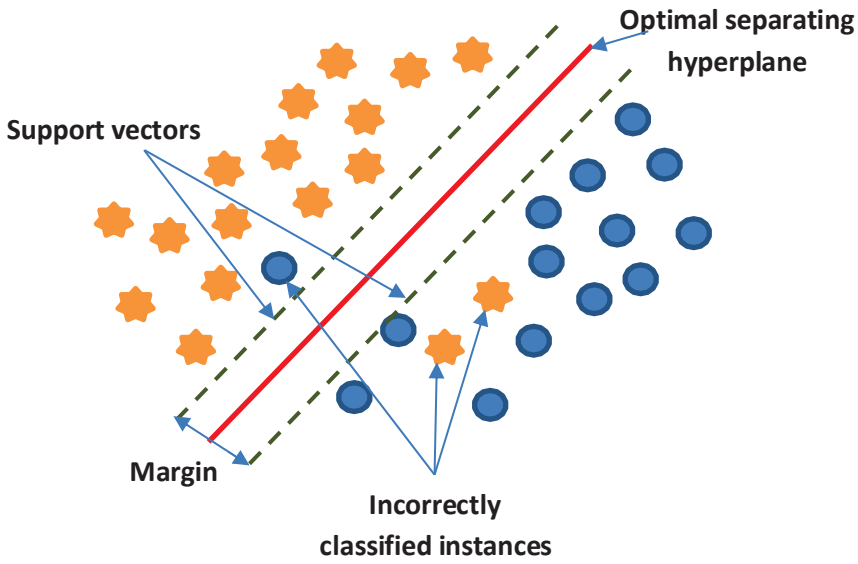
According to Breiman (2001), the processing of the RF algorithm considers two parameters: the first is the number of trees that will become a complete forest and the second refers to the number of randomly selected predictor variables, however, you must also consider a threshold value.

2.3.2 Support Vector Machine (SVM)

SVM is a machine learning method based on supervised learning. This approach is based on an optimal linear separating hyperplane, which is fitted to the training samples of two classes within a multidimensional feature space as shown in Figure 11. The optimal hyperplane is obtained by solving an optimization problem that is solved via structural risk minimization (Benediktsson and Ghamisi, 2015). The aim of the solution is to maximize the margins between the hyperplane and the closest training samples, the so-called support vectors (Vapnik, 1998). Thus, in training the classifier only samples that are close to the class boundary are needed.

Generally, the SVM is a linear classifier, i.e., a line is used to define the optimal hyperplane to discriminate the samples in the feature space, however, when a line cannot be established to discriminate the training samples, the well-known Kernel trick must be applied (Scholkopf and Smola, 2002). A kernel-based SVM is being used to project the pixel vectors into a higher dimensional space and estimate maximum margin hyperplanes in this new space, in order to improve linear separability of data (Scholkopf and Smola, 2002).

FIGURE 11 – REPRESENTATION OF SVM CLASSIFICATION



FONT: Adapted from Benediktsson and Ghamisi (2015).

3 DATA AND METHODOLOGY

This chapter is organized as follows: first, the software and hardware that were used to develop the model are described. Then, the hyperspectral data that were used in the experiments are described. After the description of the software and data, the methods are presented. It is described the methods that build up the proposed autoencoder model enhanced with convolutional layers. Then, the methods used to evaluate its applicability and performance are described.

3.1 SOFTWARE AND HARDWARE

The hardware used for the training of the proposed model is equipped with a Core I9 CPU and 16 GB in RAM, and was coded in the Python environment. The Pyzo interface (free and open-source computing environment) was used to edit and run the necessary programs. The TensorFlow and Keras libraries, included in Pyzo, were applied to develop the neural nets of the proposed SAE-1DCNN model. Although the core of the study was developed in Python, additional software (ENVI and Multispec) was used to visualize the images and help the quality assessment.

3.2 EXPERIMENTAL DATA

In order to evaluate the performance of the proposed method, four hyperspectral data sets obtained at different locations were used. The first set, Canguiri farm was obtained by the author in collaboration with KIT (Karlsruhe Institute of Technology, Germany), while the other three were downloaded from well-known research institutions such as: Purdue University, USA for the Airborne Visible / Infrared Imaging Spectrometer (AVIRIS) sensor and University of Pavia, Italy (Prof. P. Gamba) for the Reflective Optics System Imaging Spectrometer (ROSIS) sensor data (Indian Pines, Salinas and Universidad of Pavia). They are listed below, together with the necessary ground truth map used to train and evaluate the classifier.

3.2.1.1 Sampling

Training and test samples for each land cover class were selected based on the available ground truth map. In our study, the element is the pixel described by its digital values in all the available hyperspectral bands. Therefore, the model is said to be 1D because it does not include regions of the image.

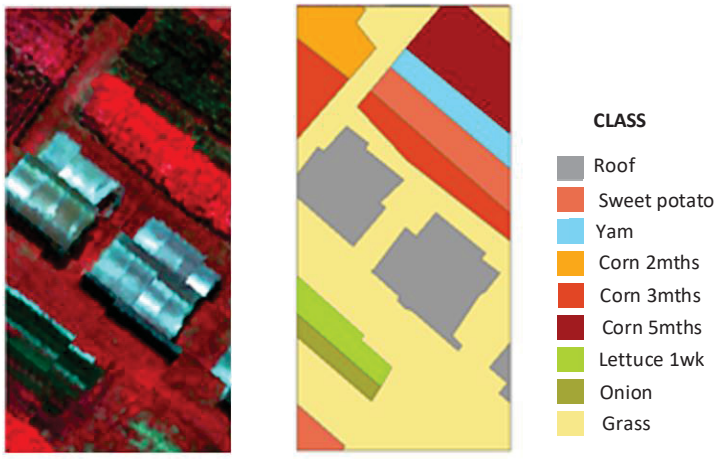
The labeled and collected samples for the three hyperspectral data sets were divided into training and test samples. Of the total samples collected, 30% were used as test samples (learning the weights and biases of each neuron) and the remaining 70% as training samples (model architecture design). Details of the samples are presented in Tables 1 – 4.

3.2.2 Canguiri Experiment

The Canguiri data set was captured in the summer season in 2020 using the Cubert's FireflEYE S185 sensor over the experimental Canguiri Farm of the Federal University of Paraná (UFPR) in Paraná-Brazil (Figure 12a). This data collection was possible thanks to the support of KIT and the Institute of Photogrammetry and Remote Sensing (IPF) in Karlsruhe-Germany that provided the UAV (Unmanned Aerial Vehicle) equipment with the hyperspectral sensor to collect the data.

The surveyed area includes some buildings but also regular shaped thin fields of different crops with similar spectral signatures (Figure 12b). The hyperspectral image size is 249 pixels × 124 pixels with spatial resolution of 0.30 m and 138 bands in the range of 0.450 – 0.998 μm with average interval between spectral bands around 4 nm. After removing 13 noisy bands, 125 bands were available for the experiment. This image has been calibrated and radiometrically corrected using a Spectralon. Table 1 lists the training and test samples collected based on the available ground truth map. The classes that were considered are: Roofs, Sweet potato, Yam crop, corn at with different growing seasons (2, 3 and 5 months), Lettuce 1wk (1 week), Onion and Grass. It should be mentioned that the shadow class was not considered in this scene.

FIGURE 12 - (A) HYPERSPECTRAL IMAGE OF THE CANGUIRI FARM DATA SET. (B) GROUND TRUTH CLASSIFICATION MAP OF CANGUIRI FARM DATA SET



(a) (b)

FONT: The author (2021)

TABLE 1 - INFORMATION CLASSES AND TRAINING-TEST SAMPLES FOR THE CANGUIRI FARM DATA SET

Class	Training	Test	Total
Roofs	273	119	392
Sweet potato	174	71	245
Yam	204	90	294
Corn 2mths	157	88	245
Corn 3mths	244	99	343
Corn 5mths	279	113	392
Lettuce 1wk	145	51	196
Onion	139	57	196
Grass	305	136	441

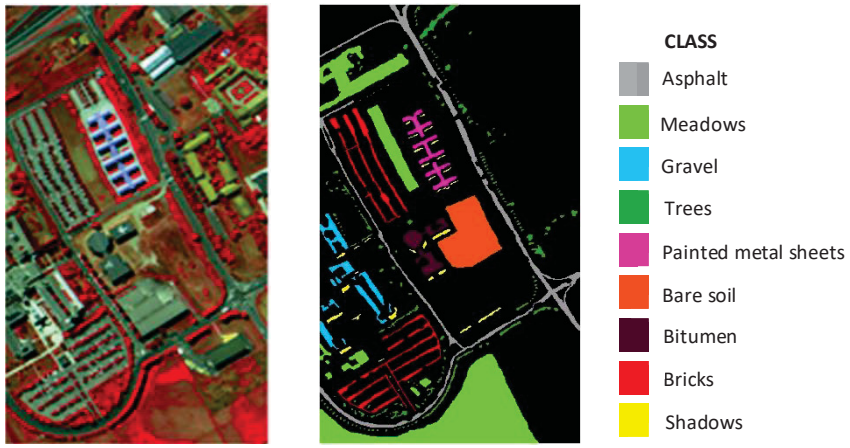
FONT: The author (2021).

3.2.3 Pavia Experiment

The second data set was recorded by the ROSIS airborne sensor during a flight campaign over the campus of the University of Pavia, northern Italy (Figure 13a). This scene considered 103 spectral bands covering the range of 0.43 to 0.86 μm with a spectral halfwidth of 7.6 nm. The University of Pavia image was radiometrically and spatially calibrated to: derive physical parameters from the measured radiance. The image consists of 610 x 340 pixels with spatial resolution of

1.3 m and contains nine different associated classes, listed in Table 2. The land cover samples (training and test) for each class were selected from the available ground truth map (Figure 13b) and are displayed in Table 2. This data set is available for download for scientific purposes at the University of Pavia repository.

FIGURE 13 - (A) HYPERSPECTRAL IMAGE OF THE UNIVERSITY OF PAVIA DATA SET. (B) GROUND TRUTH CLASSIFICATION MAP OF THE UNIVERSITY OF PAVIA DATA SET



(a) (b)

FONT: Adapted from University of Pavia repository

TABLE 2 - INFORMATION CLASSES AND TRAINING-TEST SAMPLES FOR THE UNIVERSITY OF PAVIA DATA SET

Class	Training	Test	Total
Asphalt	4623	2008	6631
Meadows	13073	5576	18649
Gravel	1465	634	2099
Trees	2149	915	3064
Painted metal sheets	962	383	1345
Bare soil	3508	1521	5029
Bitumen	940	390	1330
Bricks	2550	1132	3682
Shadows	673	274	947

FONT: The author (2021).

3.2.4 Indian Pines Experiment

The Indian Pines data set was captured by the AVIRIS airborne sensor in Northwest Indiana in the United States of America (Figure 14a). This area is covered

by mixed agricultural fields that are difficult to classify. The image size is 145 × 145 pixels and the spatial resolution is 20 m. This image has been calibrated and radiometrically corrected. For this scene, 34 spectral bands due to water absorption and noise were discarded. Therefore, 190 bands covering the range of 0.4 to 2.5 μm wavelength with an average spectral bandwidth of 10 nm were used for the experiments. The ground truth available in Figure 14b is divided into seven classes (Table 3), considering training and test samples of each class that were distributed throughout the full scene for land classification. The AVIRIS sensor hyperspectral dataset was provided by the University of Purdue-USA.

FIGURE 14 - (A) HYPERSPECTRAL IMAGE OF INDIAN PINES DATA SET. (B) GROUND TRUTH CLASSIFICATION MAP OF INDIAN PINES DATA SET

(a) (b)

FONT: Adapted from University of Purdue repository

TABLE 3 - INFORMATION CLASSES AND TRAINING-TEST SAMPLES FOR THE INDIAN PINES DATA SET

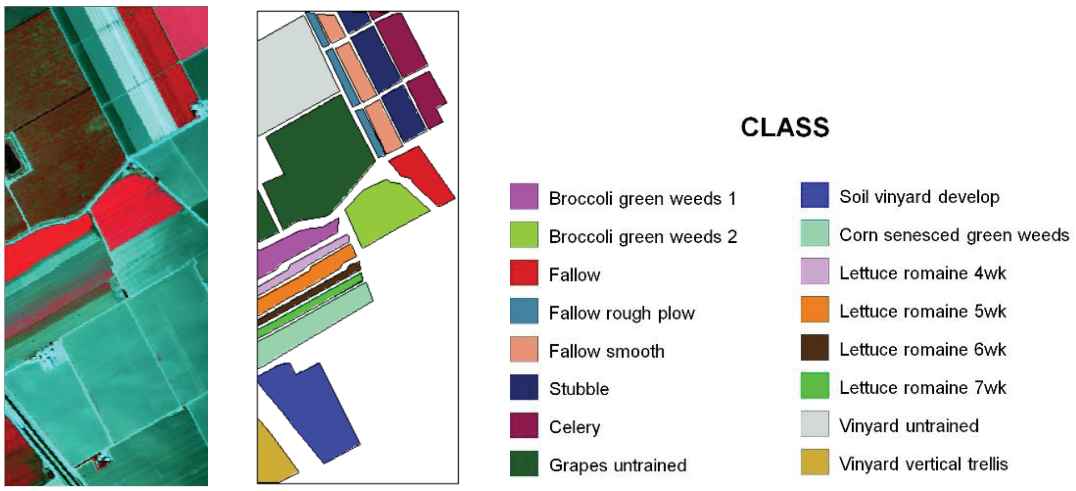
Class	Training	Test	Total
Vegetation	205	89	294
Corn-no till	266	127	393
Corn-min	232	111	343
Grass/Pasture	101	47	148
Grass/trees	238	106	344
Soybeans-no till	315	104	419
Soybeans-min	448	190	638

FONT: The author (2021).

3.2.5 Salinas Experiment

The last hyperspectral data set used in this study is the Salinas image (Figure 15a). This hyperspectral image was collected over the Salinas Valley, California, recording 224 spectral bands covering the range of 0.4 to 2.5 μm with 512 x 217 pixels through the AVIRIS airborne sensor. Like the Indian Pines image, the Salinas image has been calibrated and radiometrically corrected. The spatial resolution of this scene is 3.7 m per pixel. As mentioned above, 20 noise bands must be discarded in the AVIRIS sensor due to water vapor absorption, so only 204 bands were used for further analysis. Figure 15b shows the ground truth of this scene, which was divided into 16 classes of land cover and its training and test samples are presented in Table 4.

FIGURE 15 - (A) HYPERSPECTRAL IMAGE OF THE SALINAS DATA SET. (B) GROUND TRUTH CLASSIFICATION MAP OF SALINAS DATA SET



(a) *(b)*

TABLE 4 - INFORMATION CLASSES AND TRAINING-TEST SAMPLES FOR THE SALINAS DATA SET

Class	Training	Test	Total
Broccoli green weeds 1	1401	608	2009
Broccoli green weeds 2	2596	1130	3726
Fallow	1358	618	1976
Fallow rough plow	963	431	1394
Fallow smooth	1867	811	2678

TABLE 4 - INFORMATION CLASSES AND TRAINING-TEST SAMPLES FOR THE SALINAS DATA SET

(continuation)

Class	Training	Test	Total
Stubble	2754	1205	3959
Celery	2447	1132	3579
Grapes untrained	7963	3308	11271
Soil vinyard develop	4373	1830	6203
Corn senesced green weeds	2275	1003	3278
Lettuce_romaine_4wk	740	328	1068
Lettuce_romaine_5wk	1355	572	1927
Lettuce_romaine_6wk	655	261	916
Lettuce_romaine_7wk	746	324	1070
Vinyard untrained	5127	2141	7268
Vinyard vertical trellis	1270	537	1807

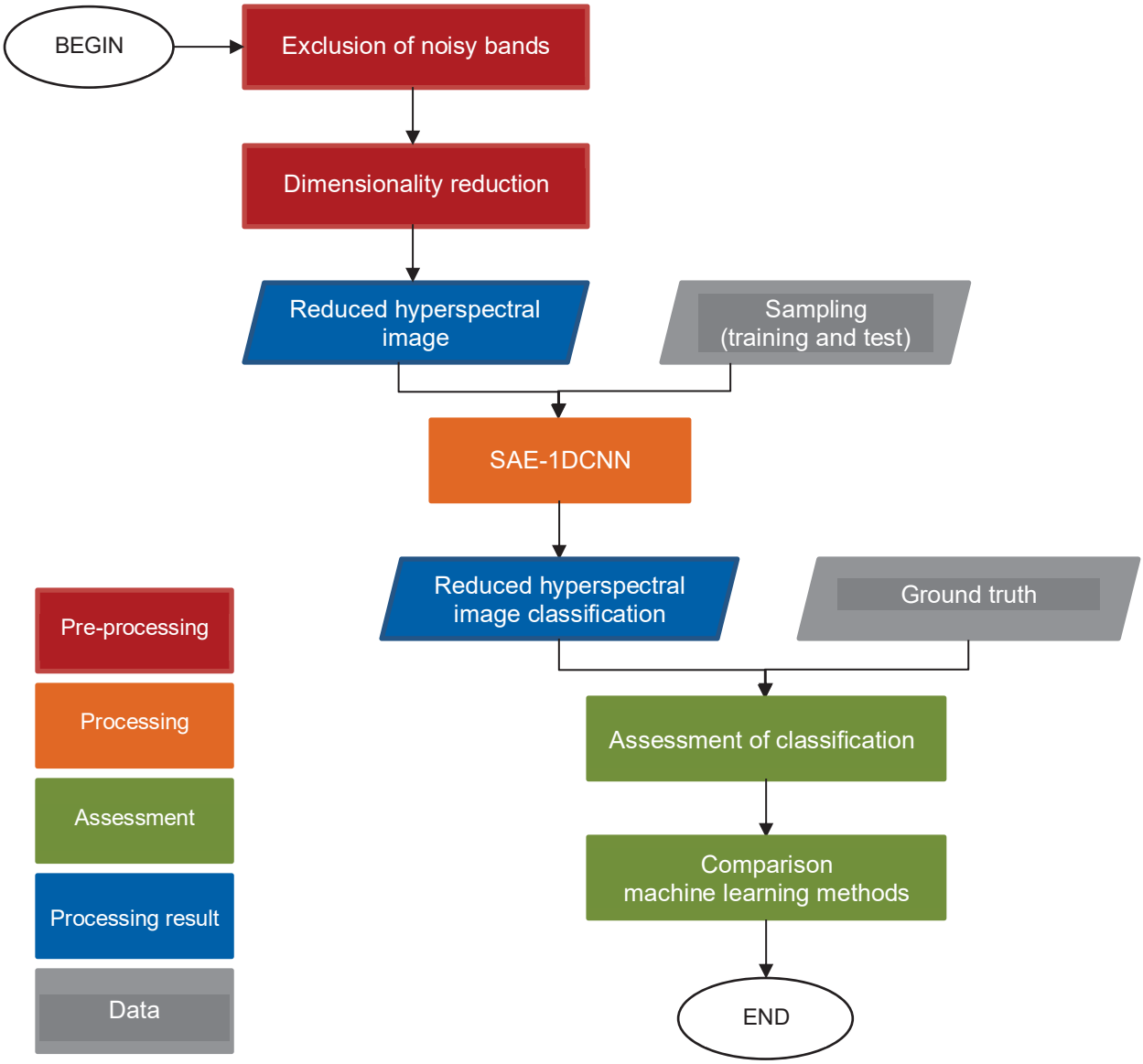
FONT: The author (2021).

3.3 METHODOLOGY

Figure 16 shows the proposed methodology. First, noisy bands, caused by the presence of atmospheric water vapor, were excluded from each data set. In the second step, dimensionality reduction was performed to reduce the spectral redundancy of the hyperspectral data while preserving the information to distinguish between different land cover classes. This first two steps can be understood as pre-processing steps: exclusion of noisy bands and dimensionality reduction. They are common in hyperspectral remote sensing image classification.

After the pre-processing steps, training and test samples were collected to perform the classification with the SAE-1DCNN model. The classification is composed by two parts: First, the SAE-1DCNN model is used to perform an unsupervised classification using the training samples without their labels. This allowed computing relevant features from the available data. Then, a supervised training is performed, including labelled samples. Here, the SAE-1DCNN parameters are adjusted based on the labels of the training samples (supervised fine-tuning). Finally, the classification obtained with the SAE-1DCNN approach were also compared to those obtained using other machine learning methods, such as SVM, ANN, CNN and SAE.

FIGURE 16 - FRAMEWORK OF THE STEPS OF THE METHODOLOGY



FONT: The author (2021)

3.3.1 Hyperspectral Data Pre-processing

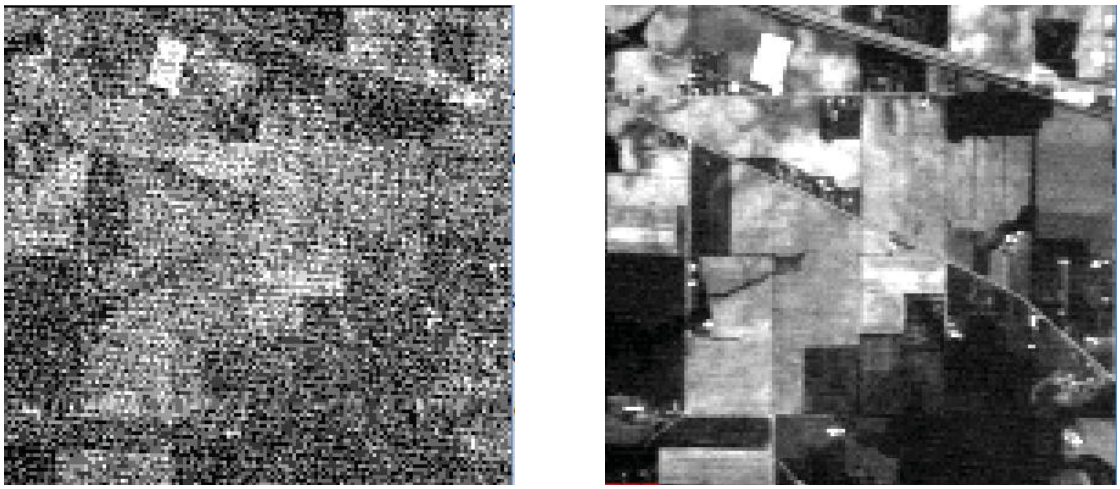
Electromagnetic radiation can be strongly absorbed by water vapor in the atmosphere, which prevents remote sensing in specific spectral regions. As hyperspectral sensors collect data along a broad spectral range, including water absorption bands in the middle infrared, it is necessary to exclude such bands to reduce noise. Therefore, an initial pre-processing step, aimed at excluding noisy bands is necessary. Even after excluding noisy bands, the data set is still large, with high redundancy. Therefore, a common practice is to reduce the number of bands

and reduce the redundancy applying feature selection or feature extraction algorithms, which constitutes a second pre-processing step. These two steps are discussed below.

3.3.1.1 Exclusion of noisy bands

In this study, noisy bands were eliminated based on two approaches. Initially, noisy bands were detected and eliminated by visual inspection (traditional and subjective approach). For this purpose, each band of each hyperspectral set was displayed on the screen where its noise was visually evaluated, considering the exaggerated roughness produced by noise. To illustrate the effect of noise, a noisy band with water vapor interference and band without noise from the Indian Pines hyperspectral image of the AVIRIS sensor are displayed in Figure 17. This is a simple method but is also affected by the user's experience, which can lead to different results.

FIGURE 17 - VISUALIZATION OF BANDS FROM THE INDIAN PINES HYPERSPECTRAL IMAGE
(a) NOISY BAND (BAND 103 = 1352,68 nm), (b) BAND WITHOUT NOISE (BAND 120 = 1620,98 nm)



(a) (b)

FONT: The author (2021)

3.3.1.2 Dimensionality reduction

Although it is expected that increasing the number of bands would increase spectral information, it has been proven that using a large number of spectral bands reduces the accuracy of the classification due to the Hughes phenomenon (Hughes, 1968). Therefore, dimensionality reduction is an important step (Zabalza et al., 2016). Dimensionality reduction methods can be divided into feature extraction and feature selection (Serpico et al., 2003). In this study, the dimensionality reduction was performed applying a machine learning algorithm based on the feature selection approach known as RF.

On issues of dimensionality reduction, this algorithm allows the selection of significant and relevant features from a large dataset (original), preserving the original information for future analyzes (Belgiu and Drăguţ, 2016). In this context, RF consists of a combination of tree-structured classifiers and can be defined according to Equation 10.

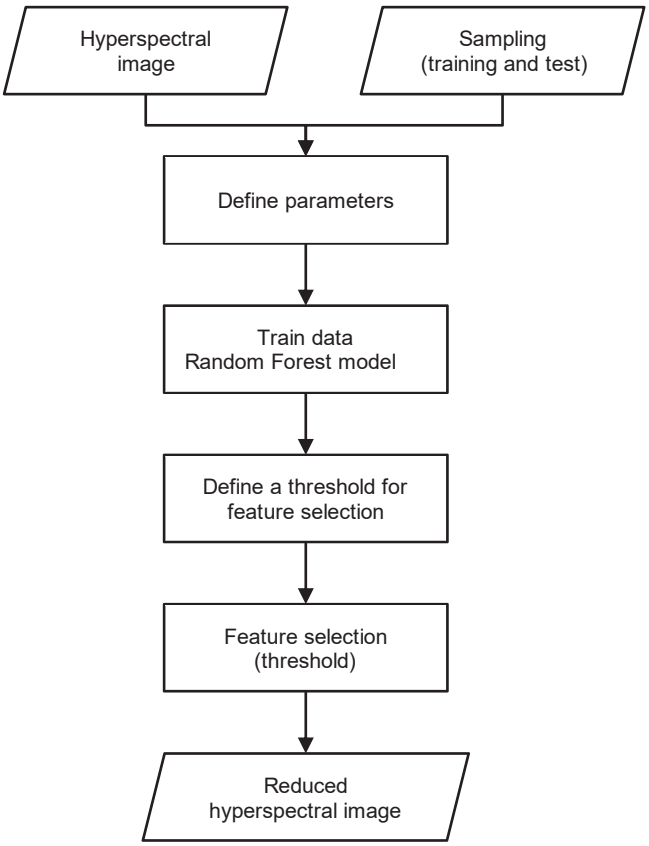
$$\{h(x, \Theta_k), k = 1, \dots\} \tag{10}$$

where *h* represents the Random Forest classifier, x is input vector and the Θ_k are independent identically distributed random vectors.

According to Breiman (2001); Feng et al., (2015) and Belgiu and Drăgut (2016), the algorithm creates multiple classification and regression trees, each trained on a bootstrapped sample of the original training data and searches only across a randomly selected subset of the input variables to determine a split (for each node). The feature selection is performed analyzing the output of all the involved decision trees (Gislason et al., 2006). In this context, the RF algorithm works from the input (labeled) samples, where, multiple decision trees are created, each with a subset of randomly chosen variables.

In order to reduce the spectral dimension of the hyperspectral images using the RF algorithm (coded in the Python environment), the parameters (number of trees and number of randomly selected predictor variables) were specified and the threshold for the selection of representative bands was defined. Figure 18 shows the methodological procedure of the RF algorithm.

FIGURE 18 - FLOWCHART OF RANDOM FOREST MODEL



FONT: The author (2021).

3.3.2 Hyperspectral Data Processing

3.3.2.1 SAE-1DCNN model

A classical AutoEncoder (AE) computes a reduced set of new variables in the hidden layer as the weighted addition of the input variables, according to equation 6. Traditionally, the input of the autoencoder is the original data set, the digital values in all available spectral bands, which is processed in the hidden layers. In the hidden layers, the input variables (digital values in different spectral bands) are combined applying weights to each input variable, and the output is the result of the weighted sum. This approach is modified in the present work. Instead of computing the weighted sum of the digital values in all spectral channels, the output is computed using only the information of spectrally neighboring channels, based on spectral regions. So, local spectral patterns are detected, and these patterns can be used to compute features that can help summarize the spectral signature of each pixel. To

compute the output, several one-dimensional convolutional nets are applied. The intention is to concentrate on the spectral relationship between neighboring bands instead of the spatial relationship between neighboring pixels, as it is traditionally done when CNN is applied.

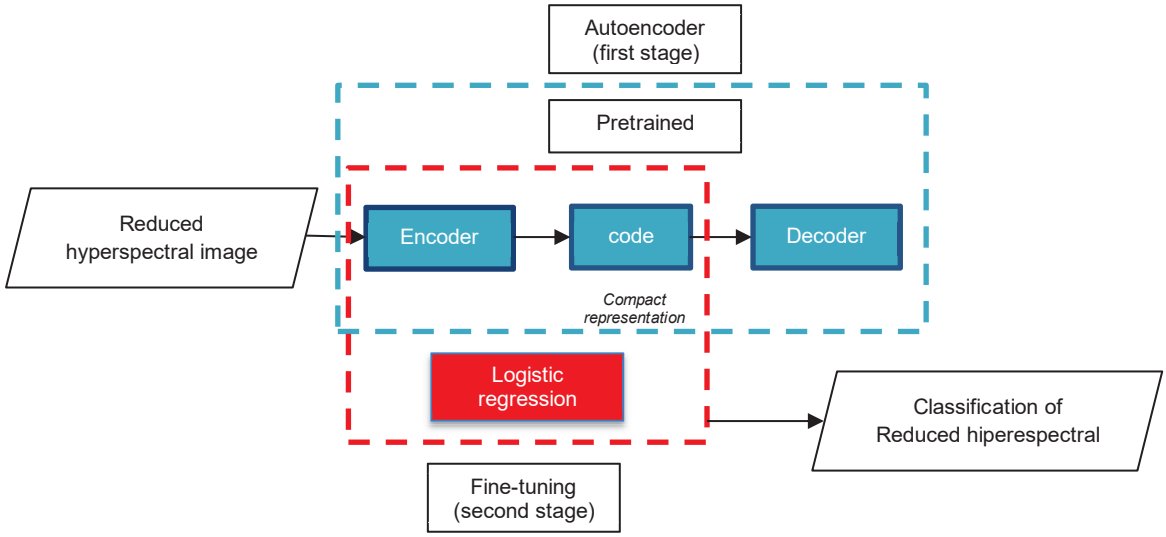
This is possible considering that the set of hyperspectral measurements of a pixel resembles a continuous series, its spectral signature, which is characterized by local spectral variations that can be detected by a CNN. In such series, spectral signature, adjacent spectral bands are highly correlated and local variations are caused by the presence of specific elements, like water, chlorophyl or iron, for example, which introduce variations that are relative smooth. So, the complete spectral set of digital values of one pixel can also be understood as a one-dimensional function, in discrete form.

As the central idea is to replace the weighted sum of the spectral data by a convolutional net, a new value is computed from the input data applying the convolution concept to the spectral series. This is equivalent to say that the spectral series is “filtered” using several 1D linear filters to compute the output values. The use of one-dimensional filters has the advantage that they are faster to compute than the 2D filters used in traditional convolutional layers. This significantly reduces the processing time.

The use of CNN enables describing the input vector with a reduced set of features, computed from spectral neighboring bands, enhancing the shape of the signatures.

Two different deep learning models are used in the proposed SAE-1DCNN classifier. The first one builds up the shell of the system and is a combination of multiple layers of Autoencoders (Stacked AutoEncoders - SAE), which is finally adapted to perform classification in a tuning step based on logistic regression. This architecture is improved by inserting a second model based on convolutional Neural Nets CNN in the encoding and decoding steps. Figure 19 displays the mechanism of the proposed model based on the two stages mentioned above. In the following subsections it is described the principles of the use of stacked autoencoders and fine tuning and the refinement of this concept by the introduction of the convolutional layers.

FIGURE 19 - GLOBAL TRAINING MECHANISM OF THE PROPOSED MODEL (SAE-1DCNN)



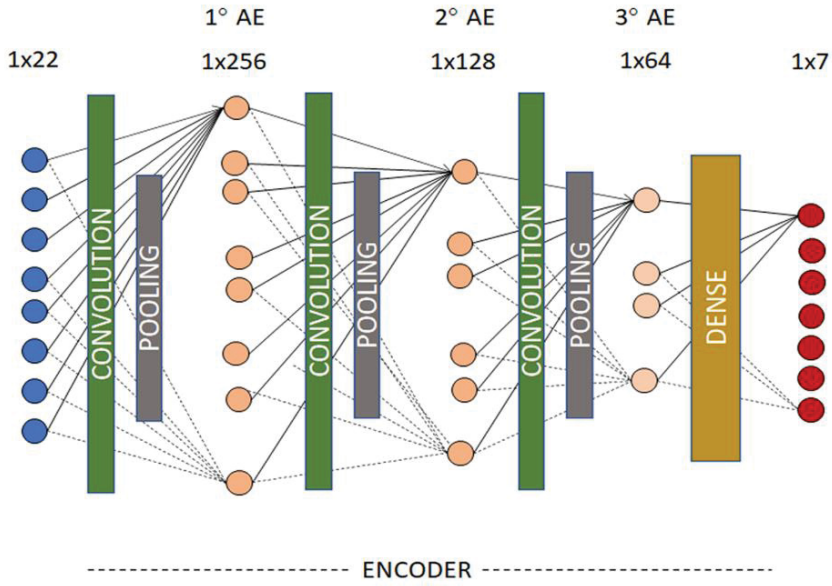
FONT: The author (2021).

3.3.2.1.1 The SAE-1DCNN Architecture

The architecture of the proposed SAE will be described using the encoding phase, with the help of Figure 20 (Indian Pines data set). As the net is symmetric, the decoding phase is symmetric to the encoding one.

The size of the input layer is equal to the number of previously selected bands. For example, 22 bands for the Indian Pines data set. Then, the data is feed into a series of AE that use three convolutional layers (reducing their size progressively). In each convolutional step, the input data is filtered using 1x3 linear filters, and the ReLU activation function is used to compute the output. Then, the result is down sampled with a 1x3 kernel and stored in a pooling layer. In the first AE, it is used 256 filters. The second uses 128 and the last one 64. The output of this net is then used as input of a conventional (dense) neural net to compute a reduced number of neurons in the latent vector. The size of the latent vector was set equal to the number of desired features. For the Indian Pines scene, the latent vector size was seven.

FIGURE 20 - PROPOSED SAE ARCHITECTURE-ENCODING PHASE



FONT: The author (2021).

The SAE-1DCNN network is first used to extract relevant features from the samples and a reduced representation of the original data set can be found in the center of the net. So, for the next step, the weights that are obtained in this step are used as start point for the next step: fine-tuning.

Fine-tuning consists of adjusting the weights for a desired purpose. In our case, the classification of the samples in the desired classes. The Fine-tuning step uses two fully connected layers. The output from the previous encoding stage is used as input in the first fully connected layer and passed to the second fully connected layer, with less neurons, that computes the final output (classification) using a conventional, dense, layer and the logistic regression based on the Softmax activation function. As the proposed model is trained to classify pixels according to their digital values in a series of hyperspectral bands, the number of neurons in the output layer is equal to the number of classes.

The parameters of the proposed SAE-1DCNN model used to process three scenes are detailed in Table 5. This table includes the number of iterations, number of hidden units, learning rate, batch size, the depth of the network, number of AEs, etc., for both the pre-training (SAE-encoder) and the supervised fine-tuning stage.

TABLE 5 - INFORMATION OF THE ARCHITECTURE OF THE SAE-1DCNN MODEL FOR HYPERSPECTRAL DATA SET

SAE-1DCNN proposed method				
Hyperspectral data set	**Parameters**			
	Pretraining (encoder)		**Fine-tuning**	
	Parameter's stage	**Filters**	**Parameter's stage**	**Full Connected Neurons**
Canguiri Farm	Epoch=50; Optimizer=Adam; Batch size=64	256 128 128 128 64	Epoch=100; Optimizer=Adam; Learn rate=0.001	300 300 9
University of Paiva	Epoch=100; Optimizer=Adam; Batch size=32	256 128 64	Epoch=300; Optimizer=Adam; Learn rate=0.001	300 100 9
Indian Pines	Epoch=50; Optimizer=Adam; Batch size=32	256 128 64	Epoch=1500; Optimizer=Adam; Learn rate=0.001	3000 100 7
Salinas	Epoch=300; Optimizer=Adam; Batch size=32	256 128 64	Epoch=300; Optimizer=Adam; Learn rate=0.001	300 300 16

FONT: The author (2021).

3.3.2.2 Unsupervised classification with Autoencoders

The basic idea of this stage is to perform an unsupervised training with stacked autoencoders that include convolutional and pooling layers that belong to the CNN concept. To carry out this procedure, this network was based on equations 6 and 7. As mentioned earlier, the encoder stage compacts the input data to a relatively short representation and decoder one reconstructs the original input from that short representation minimizing loss of information.

The parameters for the training of this model are mentioned in section 3.2.2.1.1 in Table 5. Once the unsupervised training step the proposed model (SAE-1DCNN) has been carried out, Fine-tuning is applied to adjust the network parameters according to labelled training samples, which allows performing land cover classification.

3.3.2.3 Supervised Fine-Tuning and Classification

Supervised fine-tuning is a procedure based on the transfer learning concept (Bengio, 2012; Donahue et al., 2014) that performs parameter adjustment (weights of

the model) of a pre-trained network by logistic regression (softmax) aiming at minimizing classification errors (Xing et al., 2016, Nogueira et al., 2017). In this study, the supervised fine-tuning approach is used to refine the parameters of the SAE model to label the results of the unsupervised classification. This process aims at refining the parameters of the hidden layers of the network (encoding and decoding) applying the logistic regression method by softmax for land cover classification purposes. This means that the pre-trained weights are used as initial weights in a new training step (fine-tuning by logistic regression) to obtain the labeled pixels.

3.3.3 Performance assessment

After the hyperspectral image classification, it is important and necessary to evaluate the quality of the results of the output classification map derived from the application of the proposed model (SAE-1DCNN). Through the evaluation of the results, the success of the classification can be measured and the confidence in the results can be estimated, indicating whether the aims of the analysis were achieved or not (Richards, 2013).

To assess the quality of the thematic maps, the confusion matrix of selected test samples with the known ground truth was analyzed. In addition, the method presented in this study was compared to traditional machine learning methods such as SVM, ANN, CNN and SAE. Producer's Accuracy (PA), User's Accuracy (UA), Overall Accuracy (OA), Average Accuracy (AA) and Kappa coefficient were used to compare the performance between methods and evaluate the accuracy of the land cover classification of each reduced hyperspectral image.

For the computation of such indexes, it is necessary to compute the confusion matrix of the labeled samples, a matrix that stores a comparison between the true and the obtained classification of each sample. The cells of the confusion matrix store the number of pixels of every class classified in the different classes. Each line represents how the samples of a given class are classified. The correct classified samples are stored in the diagonal of the confusion matrix, as displayed in Table 6.

TABLE 6 - EXAMPLE OF CONFUSION MATRIX CONSIDERING THREE CLASSES

	Classification data					
Ground truth	Class	C_1	C_2	C_3	Row total	Producer's Accuracy
	C_1	C_{11}	C_{12}	C_{13}	$\sum_{i}^{N_c} C_{1i}$	$\frac{C_{11}}{\sum_{i}^{N_c} C_{1i}}$
	C_2	C_{21}	C_{22}	C_{23}	$\sum_{i}^{N_c} C_{2i}$	$\frac{C_{22}}{\sum_{i}^{N_c} C_{2i}}$
	C_3	C_{31}	C_{32}	C_{33}	$\sum_{i}^{N_c} C_{3i}$	$\frac{C_{33}}{\sum_{i}^{N_c} C_{3i}}$
	Column total	$\sum_{i}^{N_c} C_{i1}$	$\sum_{i}^{N_c} C_{i2}$	$\sum_{i}^{N_c} C_{i3}$	N	
	User's Accuracy	$\frac{C_{11}}{\sum_{i}^{N_c} C_{i1}}$	$\frac{C_{22}}{\sum_{i}^{N_c} C_{i2}}$	$\frac{C_{33}}{\sum_{j}^{N_c} C_{ij}}$		

FONT: Adapted from Benediktsson and Ghamisi (2015).

The Overall Accuracy (OA) is computed comparing the number of correct classified samples (sum of the diagonal) to the total number of samples (N).

$$OA = \frac{\sum_{i}^{N_c} C_{ii}}{N} * 100 \tag{11}$$

The Producer's Accuracy (PA) describes how often the samples of a given class are correctly classified and is computed as the ratio between the correct classified samples of a class divided by the number of samples of this class.

The User's Accuracy (UA) describes how often the class displayed on the map is really present on the ground, the reliability of the thematic map. It is computed dividing the number of correct classified samples of a class by the number of samples classified as this class (sum along the column).

The average accuracy (AA) is computed as the sum of the producer's accuracies of all classes divided by the number of classes (N_c).

The Kappa coefficient is a statistical measure that refers to the agreement between the final classification and the reference data (ground truth). When the

classification is not better than reference data for each sample, kappa is equal zero (null agreement) while positive values show that the classification result is agreement with a ground truth and the best results are close to one. The kappa coefficient can be calculated by the following expression:

$$\kappa = \frac{N * \sum_{i}^{N_c} C_{ii} - \sum_{i}^{N_c} (C_{i+} * C_{+i})}{N^2 - \sum_{i,j}^{N_c} (C_{i+} * C_{+i})} \quad (12)$$

where, C_{ii} the number of correct classified samples (sum of the diagonal); C_{i+} total of observations in row i (shown as the sum total of each class to the right of the matrix); C_{+i} total of observations in column i (shown as the sum total of each class at bottom of the matrix). The higher the value of the kappa coefficient, the result will be considered of better quality. According to Richards (2013), Table 7 shows the performance of the resulting classification compared with the reference data, this table was adapted from Landis and Koch (1977).

TABLE 7 - AGREEMENT BETWEEN RESULTING CLASSIFICATION AND REFERENCE DATA BY KAPPA COEFFICIENT

Kappa coefficient	Classification performance
< 0,04	Poor
0,41 < κ ≤ 0,60	Moderate
0,61 < κ ≤ 0,75	Good
0,76 < κ ≤ 0,80	Excellent
0,81 < κ ≤ 1,00	Almost Perfect

FONT: From Richards (2013).

3.3.4 Sensitivity analysis and Improvement of the pre-processing steps

Once the model is calibrated with the labelled samples it can be used to classify new samples, or the rest of the image and assess the quality of the product. However, it would be relevant to assess which input variables are really being used in the model and how each one affects the result. Within a small artificial network, it

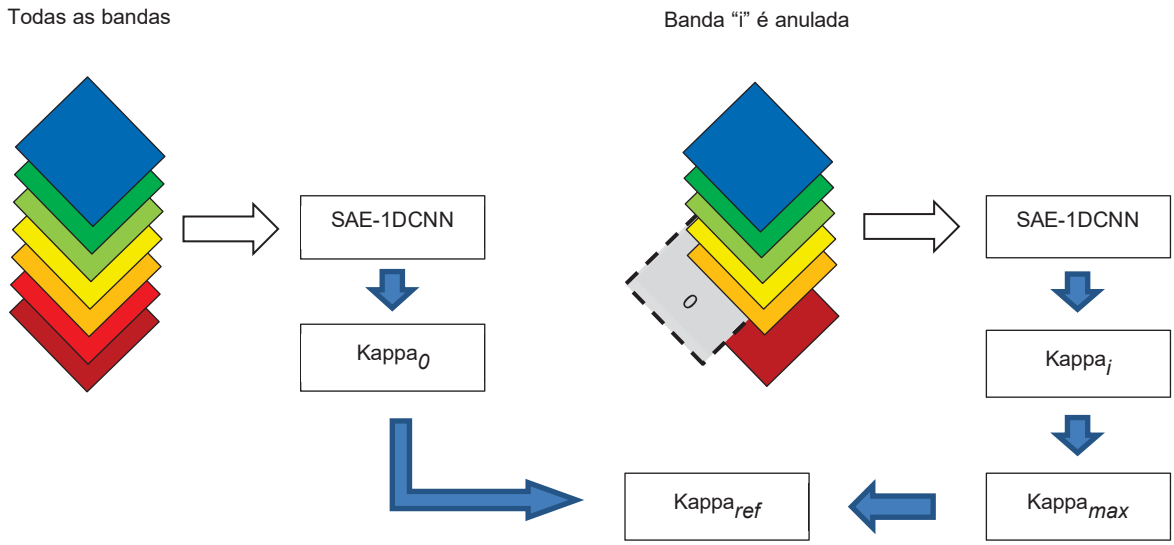
would be possible to list the weights and compute influence of each input variable, but as the SAE-1DCNN model is complex, and includes many neurons, with weights and biases, this task is difficult. Therefore, a simple approach was used to evaluate the relevance of each input variable, the sensitivity of the model to the input variables.

Sensitivity analysis of a model is a method aimed at determining how the outputs are affected based on changes in the input variables. It can be used to determine how changes in one variable affect the output. In a more general approach, the sensitivity analysis would study the extent of the changes in the output variables based on the extent of changes in the inputs.

Here, it will be analyzed how the lack of each variable influences the resulting thematic map. A solution was proposed in which the classification was repeated with the calibrated model but including one null image each time. The idea is, if a band plays an important role in the solution, substituting such band by a constant would cause a significant decrease in the kappa value. On the other hand, if the model does not use this band, the classification would not be affected. The effect of the replacement of each band was measured by the kappa coefficient obtained from the confusion matrix.

For this purpose, the SAE-1DCNN model was first calibrated with the available band set, and the accuracy of this result assessed. The kappa coefficient was used ($kappa_0$). Then, the coefficients of the net were frozen, and the model used to classify the same samples, but in each experiment one band was excluded and replaced by a constant value. This was repeated for each spectral band. The kappa coefficient was computed for each experiment. Next, the experiments were ranked according to the kappa coefficients ($kappa_i$; i = 1, 2, 3,…, number of bands) and the one with the maximal kappa coefficient identified ($kappa_{i\,max}$). In the experiments, one can find kappa values above the first reference kappa ($kappa_0$). Therefore, the reference was adjusted using the maximal kappa computed in the series of experiments. The average between the maximal kappa coefficient ($kappa_{imax}$) and the reference kappa ($kappa_0$) was computed to obtain an adjusted reference kappa ($kappa_{ref}$) as shown in expression 13. Figure 21 shows the pre-processing methodology by the SAE-1DCNN model.

FIGURE 21 - METHODOLOGY TO IMPROVE PRE-PROCESSING STEPS BY SAE-1DCNN



FONT: The author (2021).

Then, it was computed the difference between the adjusted reference kappa and the kappa coefficient of each experiment to identify the significant bands (expression 14). For this purpose, a threshold was set, empirically. If the absolute difference lies below the threshold, then the band can be discarded.

$$kappa_{ref} = 0.5 * (kappa_{i\,max} + kappa_0) \quad (13)$$

$$Dif_i = kappa_{ref} - kappa_i \quad (14)$$

where:

i : omitted band; $kappa_{ref}$: adjusted reference kappa; $kappa_{i\,max}$: maximal kappa obtained in the experiments omitting one band; $kappa_0$: kappa coefficient of the experiment using all bands.

This procedure enables detecting the most significant input variables, those that affect more significantly the quality of the result, and the less significant, those that do not change the result. The idea is, if a band plays an important role in the solution, substituting such band by a constant would cause a significant decrease in the kappa value. On the other hand, if the model does not use this band, the

classification would not be affected. The effect of the replacement of each band was measured by the kappa coefficient obtained from the confusion matrix. A threshold was applied to detect which bands can be omitted and which are significant.

The results that were obtained with the SAE-1DCNN model encouraged to investigate its potential use to perform the pre-processing steps and substitute the previously feature selection and noisy band suppression steps.

As the proposed model used autoencoders, it was verified if the autoencoder can detect which bands do not contribute to the solution and can be discarded. This would be the case of two situations:

- if a spectral band is redundant and its information is contained in other bands.
- if a spectral band does not contain relevant information. This can happen, for example, when the band is noisy and therefore not valuable for the decision about the probable class.

So, the sensitivity analysis was performed to evaluate the potential of the SAE-1DCNN model to deal with these two problems: features selection and noise detection.

3.3.4.1 Identification of noisy bands with the SAE-1DCNN model

After the first series of classification tests with the proposed SAE-1DCNN model, it was verified if the model is also able to detect noisy bands and, so, eliminate the necessity to perform visual band selection, based on the sensitivity analysis. For the detection of noisy bands applying the proposed model (SAE-1DCNN) only the Salinas and Indian Pines images of the AVIRIS sensor were used, because in these images there are noisy bands available.

First, the SAE-1DCNN model was trained with a data set including noisy bands to obtain a reference classification of a selected set of pixels. Then, the sensitivity analysis was applied, substituting one band by a constant value, which would eliminate it in the decision. The rationale behind the idea is that, if the model computes spectral features from different spectral regions and selects the most relevant ones to perform the classification, it would not include noisy regions in the solution. The weights of such regions would be null at some point in the net to reduce the unnecessary information and improve the results.

After computing the weights that can be used to classify the hyperspectral imagen in the desired classes, the weights are defined and frozen. If the input set includes a noisy band, the system will not use it to compute the solution, as this band does not include useful information. The problem here is to detect which bands are used and which are not used, which can be solved by inspecting the difference of the kappa values.

To detect the noisy bands, a threshold was applied to the kappa difference and it was analyzed if the noisy bands are included in the set of bands below the threshold.

3.3.4.2 SAE-1DCNN band selection

Knowing the potential of the SAE-1DCNN model in classification and identification of noisy bands, it was proposed to carry out a test to evaluate the potential of the SAE-1DCNN model for feature selection and identification of redundant bands. For this purpose, the sensitivity analysis was used. The steps are the same as those described in the previous subsection “Identification of noisy bands with the SAE-1DCNN model”, but in this case the interest lies in identifying the most significant bands, those that produce a significant fall of the kappa coefficient. So, it was possible to detect and discard unnecessary bands. If the difference is low relative to the threshold, the band is not considered necessary and can be eliminated. The images used in this process were Salinas and Indian Pines to continue the previous process.

4 RESULTS AND DISCUSSION

In this chapter it is presented the results obtained applying the SAE-1DCNN model to classify four different hyperspectral data sets. This includes the pre-processing steps and the classification. It is also described the results of the application of the sensitivity analysis to detect noisy bands and to perform band selection.

4.1 HYPERSPECTRAL DATA PRE-PROCESSING

The pre-processing steps described below include the detection of noisy bands as well as the reduction of the original hyperspectral set.

To detect noisy bands, the first approach was visual inspection of each band. Then, in a second attempt, the SAE-1DCNN model was used to perform the same task.

The reduction of the input variables was first performed using the RF method. In a second experiment, the sensitivity analysis was applied to select the most relevant variables and the results obtained the two approaches were compared.

4.1.1 Exclusion of noisy bands

Noisy bands were visually identified and excluded. For this purpose, each band was visualized on the computer screen (detailed in section 3.1) and the number of noisy bands was visually estimated. The result of this step depends on the experience of the analyst and different results may be obtained by different analysts, which is a critic to this method. Nevertheless, it was used as a first approach in the experiments. In a second step, tests were also performed based on the proposed SAE-1DCNN band selection approach.

The bands that were excluded for each hyperspectral image through visual identification are presented in Table 8. For further analysis, hyperspectral images without noisy bands will be used.

TABLE 8 - NOISY BANDS REMOVED BY VISUAL IDENTIFICATION

Hyperspectral data set	Total bands	Noisy bands	Used
Canguiri Farm	138	13	125
University of Pavia	115	12	103
Indian Pines	224	34	190
Salinas	224	20	204

FONT: The author (2021).

Additionally, to confirm the robustness of the SAE-1DCNN model in terms of identification of noisy bands, tests were carried out with the AVIRIS hyperspectral images of Indian Pines and Salinas (process carried out based on a code in the Python environment). The tests consisted of applying the proposed SAE-1DCNN band selection approach using band sets with and without noisy bands.

For the tests performed in this section (using the SAE-1DCNN model), the noiseless bands used for the Salinas image were a total of 18, while 22 were used for the Indian Pines image, respectively. Five noisy bands were included in the Salinas image and 10 noisy bands in the Indian Pines image. This means that 23 bands were used for the Salinas image and 32 bands for the Indian Pines image. For the Salinas image, the noisy bands used were: 108, 111, 159, 160, 162; while for the Indian Pines image: 1, 103, 104, 107, 109, 153, 155, 157, 161, 163. The noisy bands for the two evaluated scenes were randomly collected, and belong to the red regions and infrared of the electromagnetic spectrum.

In the next step, the experiment was repeated replacing the information of one band by a constant value, which means that this band has no relevant information. Then, the quality of the thematic map of each experiment was analyzed using the overall accuracy (OA), average accuracy (AA) and kappa coefficient. The thresholds of 10% for the Salinas data set and 3% for the Indian Pines data set were used to identify the bands that do not contribute significantly to the solution. The adjusted reference kappa ($kappa_{ref}$) computed and used for the Salinas image was 88% while for the Indian Pines image it was 76%.

The evaluation and identification of noisy bands was carried out based on the difference between the adjusted reference kappa ($kappa_{ref}$) and the kappa coefficient of each experiment. These differences were compared through the threshold established for each image. Thus, if the difference between these two

kappa values is less than the set threshold, it is identified as a noisy band. Based on the results obtained in each scene, all the noisy bands used in this section were identified, therefore, they can be eliminated since they do not contribute to later analysis. Tables 9 and 10 show the results obtained in the images of Salinas and Indian Pines for the identification of noisy bands.

TABLE 9 - IDENTIFICATION OF NOISY BANDS REMOVED BY SAE-1DCNN MODEL FOR SALINAS DATA SET

Hyper Data set	Index	SAE-CNN 18 bands	SAE-CNN 23 bands (18 without noise + 5 noisy bands)	SAE-CNN 23 bands (108*)	SAE-CNN 23 bands (111*)	SAE-CNN 23 bands (159*)	SAE-CNN 23 bands (160*)	SAE-CNN 23 bands (162*)	SAE-CNN 23 bands (7**)	SAE-CNN 23 bands (11**)	SAE-CNN 23 bands (17**)	SAE-CNN 23 bands (37**)	SAE-CNN 23 bands (153**)
Salinas	OA	0.89	0.89	0.90	0.90	0.90	0.90	0.90	0.62	0.73	0.30	0.38	0.51
	AA	0.94	0.94	0.96	0.96	0.96	0.96	0.96	0.62	0.72	0.36	0.36	0.43
	Kappa	87.55	87.40	89.31	89.30	89.31	89.32	89.32	58.61	69.54	25.42	31.33	45.30

FONT: The author (2021).

* noisy bands: 108, 111, 159, 160, 162
** bands without noise 7, 11, 17, 37, 153

TABLE 10 - IDENTIFICATION OF NOISY BANDS REMOVED BY SAE-1DCNN MODEL FOR INDIAN PINES DATA SET

Hyper Data set	Index	SAE-CNN 22 bands	SAE-CNN 32 bands (22 without noise + 10 noisy bands)	SAE-CNN 32 bands (1*)	SAE-CNN 32 bands (103*)	SAE-CNN 32 bands (104*)	SAE-CNN 32 bands (107*)	SAE-CNN 32 bands (109*)	SAE-CNN 32 bands (153*)	SAE-CNN 32 bands (155*)	SAE-CNN 32 bands (157*)	SAE-CNN 32 bands (161*)	SAE-CNN 32 bands (163*)
Indian Pines	OA	0.80	0.80	0.76	0.83	0.84	0.84	0.84	0.84	0.84	0.84	0.84	0.84
	AA	0.81	0.83	0.77	0.84	0.85	0.85	0.85	0.85	0.85	0.85	0.85	0.85
	Kappa	76.04	77.31	77.02	77.02	78.49	78.40	78.45	78.17	78.35	78.40	78.49	78.44

Index	SAE-CNN 32 bands (23**)	SAE-CNN 32 bands (27**)	SAE-CNN 32 bands (169**)	SAE-CNN 32 bands (171**)
OA	0.39	0.34	0.75	0.76
AA	0.26	0.24	0.78	0.77
Kappa	23.91	23.10	67.75	69.28

FONT: The author (2021).

* noisy bands: 1, 103, 104, 107, 109, 153, 155, 157 161, 163
** bands without noise 23, 27, 169, 171

4.1.2 Dimensionality reduction

After identifying and excluding noisy bands, the next pre-processing step is dimensionality reduction. To avoid the Hughes phenomenon (Hughes, 1968) on hyperspectral images, it was necessary to reduce the spectral dimension. The approach used to reduce the dimensionality in each image was RF. Table 11 displays the number of bands before and after the dimensionality reduction process.

TABLE 11 - REDUCTION OF SPECTRAL BANDS BY RANDOM FOREST

Hyperspectral image	Total bands	Reduced	Used
Canguiri Farm	125	100	25
University of Pavia	103	83	20
Indian Pines	190	168	22
Salinas	204	186	18

FONT: The author (2021).

Additional tests were performed to verify if the SAE-1DCNN model can perform dimensionality reduction. The Salinas and Indian Pines data sets were used in the experiments (explained in the section 3.3.4.1). From each data set, a subset of bands (without noise) was randomly chosen (102 for Salinas and 100 bands for Indian Pines). These sets include redundant bands.

Again, the SAE-1DCNN model was trained using all available spectral bands and a reference classification was computed, which enabled computing the OA, AA and Kappa quality indexes. Then, the same model was applied using all the available bands but substituting the values of a chosen band by a constant. This means that the model was applied 102 and 100 times for the Salinas and Indian Pines images, respectively. The results of the metrics obtained in the tests carried out on the images of the AVIRIS sensor are presented in Tables 12 and 13.

The adjusted reference kappa computed for the Salinas image was 92% while the Indian Pines image was 95%. Then, the differences between the adjusted reference kappa and the kappa of each band with constant value were computed. And then, the threshold was set to identify redundant bands based on the kappa differences.

TABLE 12 - METRICS OF SPECTRAL BANDS FOR SALINAS DATA SET

Hyperspectral Data set	Index	SAE-CNN 102 bands	SAE-CNN B2	SAE-CNN B4	SAE-CNN B6	SAE-CNN B8	SAE-CNN B10	SAE-CNN B12	SAE-CNN B14	SAE-CNN B16	SAE-CNN B18	SAE-CNN B20
Salinas	OA	0.93	0.93	0.91	0.80	0.79	0.82	0.90	0.92	0.90	0.89	0.88
	AA	0.97	0.97	0.95	0.87	0.87	0.89	0.95	0.96	0.95	0.94	0.92
	Kappa	92.49	92.65	89.58	77.23	76.61	80.30	89.09	91.37	88.56	87.28	86.52
	Index	**SAE-CNN B22**	**SAE-CNN B24**	**SAE-CNN B26**	**SAE-CNN B28**	**SAE-CNN B30**	**SAE-CNN B32**	**SAE-CNN B34**	**SAE-CNN B36**	**SAE-CNN B38**	**SAE-CNN B40**	**SAE-CNN B42**
	OA	0.88	0.89	0.90	0.83	0.77	0.79	0.89	0.89	0.88	0.77	0.69
	AA	0.91	0.92	0.93	0.87	0.82	0.81	0.89	0.90	0.89	0.72	0.65
	Kappa	86.10	87.95	88.53	81.00	74.91	76.35	87.43	87.29	86.08	73.99	65.30
	Index	**SAE-CNN B44**	**SAE-CNN B46**	**SAE-CNN B48**	**SAE-CNN B50**	**SAE-CNN B52**	**SAE-CNN B54**	**SAE-CNN B56**	**SAE-CNN B58**	**SAE-CNN B60**	**SAE-CNN B62**	**SAE-CNN B64**
	OA	0.69	0.73	0.79	0.80	0.79	0.80	0.81	0.85	0.88	0.93	0.92
	AA	0.67	0.69	0.72	0.75	0.73	0.75	0.76	0.83	0.88	0.97	0.96
	Kappa	64.72	69.79	76.17	77.94	76.59	77.32	78.68	83.38	86.37	92.25	91.36
	Index	**SAE-CNN B66**	**SAE-CNN B68**	**SAE-CNN B70**	**SAE-CNN B72**	**SAE-CNN B74**	**SAE-CNN B76**	**SAE-CNN B78**	**SAE-CNN B80**	**SAE-CNN B82**	**SAE-CNN B84**	**SAE-CNN B86**
	OA	0.88	0.83	0.82	0.85	0.88	0.91	0.93	0.94	0.94	0.93	0.93
	AA	0.93	0.89	0.87	0.89	0.92	0.94	0.96	0.97	0.97	0.97	0.96
	Kappa	86.46	81.20	80.52	83.11	86.64	89.81	91.79	92.78	92.80	92.70	91.91
	Index	**SAE-CNN B88**	**SAE-CNN B90**	**SAE-CNN B92**	**SAE-CNN B94**	**SAE-CNN B96**	**SAE-CNN B98**	**SAE-CNN B100**	**SAE-CNN B102**	**SAE-CNN B104**	**SAE-CNN B106**	**SAE-CNN B108**
	OA	0.92	0.92	0.92	0.92	0.92	0.93	0.93	0.93	0.93	0.94	0.94
	AA	0.95	0.95	0.94	0.95	0.96	0.96	0.96	0.96	0.97	0.97	0.97
	Kappa	91.30	90.98	90.51	90.64	91.56	91.69	91.63	92.15	92.67	92.85	92.89
	Index	**SAE-CNN B110**	**SAE-CNN B112**	**SAE-CNN B114**	**SAE-CNN B116**	**SAE-CNN B118**	**SAE-CNN B120**	**SAE-CNN B122**	**SAE-CNN B124**	**SAE-CNN B126**	**SAE-CNN B128**	**SAE-CNN B130**
	OA	0.94	0.94	0.94	0.93	0.88	0.82	0.85	0.92	0.93	0.93	0.93
	AA	0.97	0.97	0.97	0.97	0.92	0.84	0.88	0.95	0.96	0.96	0.97
	Kappa	92.89	92.89	92.89	92.64	86.82	79.71	82.57	90.79	91.71	92.22	92.39

TABLA 12 - METRICS OF SPECTRAL BANDS FOR SALINAS DATA SET

(continuation)

Hyperspectral Data set	Index	SAE-CNN B132	SAE-CNN B134	SAE-CNN B136	SAE-CNN B138	SAE-CNN B140	SAE-CNN B142	SAE-CNN B144	SAE-CNN B146	SAE-CNN B148	SAE-CNN B150	SAE-CNN B152
Salinas	OA	0.93	0.93	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94
	AA	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	Kappa	92.53	92.65	92.78	92.88	92.89	92.89	92.89	92.89	92.89	92.89	92.89
	Index	**SAE-CNN B154**	**SAE-CNN B156**	**SAE-CNN B158**	**SAE-CNN B160**	**SAE-CNN B162**	**SAE-CNN B164**	**SAE-CNN B166**	**SAE-CNN B168**	**SAE-CNN B170**	**SAE-CNN B172**	**SAE-CNN B174**
	OA	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94
	AA	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	Kappa	92.89	92.89	92.89	92.89	92.89	92.88	92.88	92.88	92.89	92.89	92.89
	Index	**SAE-CNN B176**	**SAE-CNN B178**	**SAE-CNN B180**	**SAE-CNN B182**	**SAE-CNN B184**	**SAE-CNN B186**	**SAE-CNN B188**	**SAE-CNN B190**	**SAE-CNN B192**	**SAE-CNN B194**	**SAE-CNN B196**
	OA	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94
	AA	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	Kappa	92.89	92.89	92.89	92.89	92.89	92.89	92.89	92.89	92.89	92.89	92.89
	Index	**SAE-CNN B198**	**SAE-CNN B200**	**SAE-CNN B202**	**SAE-CNN B204**							
	OA	0.94	0.94	0.94	0.94							
	AA	0.97	0.97	0.97	0.97							
	Kappa	92.89	92.89	92.89	92.89							

FONT: The author (2021).

TABLE 13 - METRICS OF SPECTRAL BANDS FOR INDIAN PINES DATA SET

Hyperspectral Data set	Index	SAE-CNN 100 bands	SAE-CNN B1	SAE-CNN B3	SAE-CNN B5	SAE-CNN B7	SAE-CNN B8	SAE-CNN B9	SAE-CNN B11	SAE-CNN B13	SAE-CNN B15	SAE-CNN B17
Indian Pines	OA	0.94	0.98	0.97	0.96	0.95	0.95	0.93	0.93	0.92	0.93	0.93
	AA	0.95	0.98	0.98	0.96	0.95	0.95	0.93	0.93	0.92	0.93	0.94
	Kappa	93.06	97.27	96.85	95.36	94.34	93.64	92.02	91.23	90.16	91.18	91.63
	Index	**SAE-CNN B19**	**SAE-CNN B21**	**SAE-CNN B23**	**SAE-CNN B25**	**SAE-CNN B27**	**SAE-CNN B29**	**SAE-CNN B31**	**SAE-CNN B33**	**SAE-CNN B35**	**SAE-CNN B37**	**SAE-CNN B39**
	OA	0.92	0.88	0.84	0.81	0.73	0.72	0.68	0.83	0.91	0.90	0.93
	AA	0.93	0.87	0.83	0.79	0.7	0.68	0.64	0.82	0.92	0.92	0.93
	Kappa	90.64	85.82	80.96	77.43	68.37	66.89	62.65	80.28	88.85	87.85	91.24
	Index	**SAE-CNN B41**	**SAE-CNN B43**	**SAE-CNN B45**	**SAE-CNN B47**	**SAE-CNN B49**	**SAE-CNN B50**	**SAE-CNN B51**	**SAE-CNN B53**	**SAE-CNN B55**	**SAE-CNN B57**	**SAE-CNN B59**
	OA	0.91	0.92	0.93	0.92	0.94	0.95	0.96	0.96	0.96	0.97	0.97
	AA	0.92	0.93	0.94	0.94	0.95	0.96	0.96	0.97	0.97	0.98	0.97
	Kappa	89.63	90.47	91.89	90.80	92.68	94.28	95.01	95.56	95.55	96.94	96.34
	Index	**SAE-CNN B61**	**SAE-CNN B63**	**SAE-CNN B65**	**SAE-CNN B67**	**SAE-CNN B69**	**SAE-CNN B70**	**SAE-CNN B71**	**SAE-CNN B73**	**SAE-CNN B75**	**SAE-CNN B77**	**SAE-CNN B79**
	OA	0.97	0.97	0.97	0.96	0.94	0.92	0.92	0.96	0.98	0.98	0.98
	AA	0.97	0.97	0.97	0.97	0.93	0.87	0.87	0.96	0.98	0.98	0.98
	Kappa	95.83	96.24	96.20	95.69	93.40	90.60	90.41	95.27	97.36	97.41	97.13
	Index	**SAE-CNN B81**	**SAE-CNN B83**	**SAE-CNN B85**	**SAE-CNN B87**	**SAE-CNN B89**	**SAE-CNN B91**	**SAE-CNN B93**	**SAE-CNN B95**	**SAE-CNN B97**	**SAE-CNN B99**	**SAE-CNN B101**
	OA	0.94	0.93	0.94	0.95	0.95	0.96	0.96	0.95	0.97	0.98	0.98
	AA	0.95	0.94	0.95	0.96	0.96	0.96	0.96	0.96	0.98	0.98	0.98
	Kappa	93.37	91.66	93.05	93.84	94.58	94.76	94.62	93.98	96.57	97.45	97.36
	Index	**SAE-CNN B103**	**SAE-CNN B105**	**SAE-CNN B107**	**SAE-CNN B109**	**SAE-CNN B110**	**SAE-CNN B111**	**SAE-CNN B113**	**SAE-CNN B115**	**SAE-CNN B117**	**SAE-CNN B119**	**SAE-CNN B121**
	OA	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
	AA	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
	Kappa	97.36	97.36	97.36	97.41	97.50	97.45	97.45	97.50	97.45	97.41	97.41

TABLE 13 - METRICS OF SPECTRAL BANDS FOR INDIAN PINES DATA SET

(continuation)

Hyperspectral Data set	Index	SAE-CNN B123	SAE-CNN B125	SAE-CNN B127	SAE-CNN B129	SAE-CNN B131	SAE-CNN B133	SAE-CNN B135	SAE-CNN B137	SAE-CNN B139	SAE-CNN B141	SAE-CNN B143
Indian Pines	OA	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
	AA	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
	Kappa	97.41	97.41	97.36	97.36	97.36	97.36	97.36	97.36	97.36	97.36	97.36
	Index	**SAE-CNN B145**	**SAE-CNN B147**	**SAE-CNN B149**	**SAE-CNN B150**	**SAE-CNN B151**	**SAE-CNN B153**	**SAE-CNN B155**	**SAE-CNN B157**	**SAE-CNN B159**	**SAE-CNN B161**	**SAE-CNN B163**
	OA	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
	AA	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
	Kappa	97.36	97.36	97.36	97.36	97.36	97.36	97.36	97.36	97.36	97.36	97.36
	Index	**SAE-CNN B165**	**SAE-CNN B167**	**SAE-CNN B169**	**SAE-CNN B171**	**SAE-CNN B173**	**SAE-CNN B175**	**SAE-CNN B177**	**SAE-CNN B179**	**SAE-CNN B181**	**SAE-CNN B183**	**SAE-CNN B185**
	OA	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
	AA	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
	Kappa	97.36	97.36	97.36	97.36	97.36	97.36	97.36	97.36	97.36	97.36	97.36
	Index	**SAE-CNN B187**	**SAE-CNN B189**									
	OA	0.98	0.98									
	AA	0.98	0.98									
	Kappa	97.36	97.36									

FONT: The author (2021).

Analyzing the metrics obtained for each test, it was possible to identify the bands that contribute to the land cover classification on each scene. For example, in the Salinas image (see Graphic 1), 18 useful bands were identified, being 1 in the blue region, 2 in the green region, 3 in the red region, 11 in the near infrared region and 1 in the middle infrared, while for the Indian Pines image (see Graphic 2), 22 bands were identified, 5 belong to the green region, 8 to the red region and 9 to the near infrared. This procedure enables detecting the most useful bands and allows eliminating those that are less significant (Table 14).

TABLE 14 - REDUCTION OF SPECTRAL BANDS BY SAE-1DCNN MODEL

Hyperspectral image	Total bands	Reduced	Used
Salinas	102	84	18
Indian Pines	100	78	22

FONT: The author (2021).

To evaluate the set of selected bands, they were used to classify the images, and quality indexes computed. The land cover classification of each scene (Salinas and Indian Pines) was performed using three methods, the first was SVM, the second was CNN and the third was the same proposed SAE-1DCNN approach. The results obtained based on these three classification methods gave accurate values in their metrics (Table 15), indicating that the proposed method can be an alternative to reduce the spectral dimension of hyperspectral images.

TABLE 15 - METRICS OF LAND COVER CLASSIFICATION FOR REDUCED HYPERSPECTRAL DATA SETS

Hyperspectral Data set	Index	SVM	CNN	SAE-CNN
Salinas	OA	0.93	0.92	0.92
	Kappa	91.71	91.36	91.40
Indian Pines	**Index**	**SVM**	**CNN**	**SAE-CNN**
	OA	0.91	0.80	0.92
	Kappa	89.84	76.13	90.60

FONT: The author (2021).

Additionally, a comparison was made between the classification metrics (kappa coefficient) obtained using RF and SAE-1DCNN as dimensionality reduction methods (Table 16). For example, for the Indian Pines image, applying the RF method for dimensionality reduction and the SVM method for land cover classification, a kappa value of 71.36% was obtained, while using the SAE-1DCNN method for dimensionality reduction and the SVM for classification was obtained 89.84%. Another example can be identified when applying the RF method for dimensionality reduction and for land cover classification the proposed method (SAE-1DCNN), the kappa coefficient gave a value of 76.04%, while applying the SAE-1DCNN model for reduction of dimensionality and also for land cover classification obtained a kappa value of 90.60%. With these data mentioned above, it is possible to identify the effectiveness of the proposed model (SAE-1DCNN) as a dimensionality reduction model, because when choosing the appropriate method for dimensionality reduction the most representative bands in a hyperspectral image are selected and thus, it increases the land cover classification accuracy.

TABLE 16 - KAPPA COEFFICIENT COMPARISON USING RF AND SAE-1DCNN AS DIMENSIONALITY REDUCTION METHODS FOR INDIAN PINES DATA.

Hyperspectral Data set	Dimensionality reduction method	Index	SVM	CNN	SAE-CNN
Indian Pines	Random Forest	Kappa	71.36	65.71	76.04
	SAE-1DCNN		89.84	76.13	90.60

FONT: The author (2021).

Also, a comparison was made between the bands selected by the RF method and the SAE-1DCNN method for dimensionality reduction (Graphic 3 and Graphic 4) for the hyperspectral images Salinas and Indian Pines.

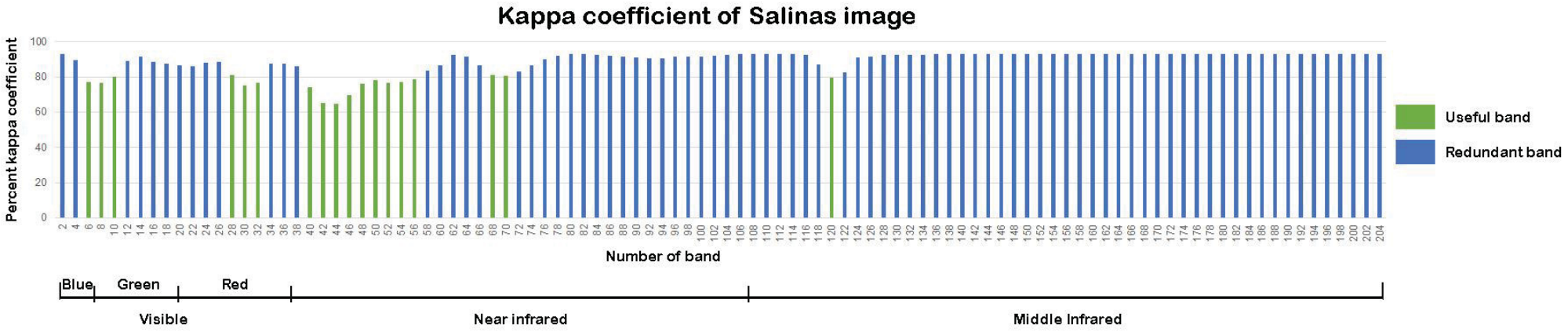
Using the Salinas data set, the proposed method selects several bands in the near infrared region (Graphic 3), where the reflectance of vegetation is high, which allows discriminating the different crops that are represented in this scene. On the other hand, the RF method selects mostly bands in the visible portion of the electromagnetic spectrum. This can be good to discriminate impervious areas but is not good to classify different crops.

Similarly, when the Indian Pines image is used, the SAE-1DCNN method selects several bands in the near infrared, while the Random Forest method recommends bands in the middle-infrared (Graphic 4), which contributes to the discrimination of vegetation classes based on these two methods, however, it must be taken into account that for this scene the proposed model selected bands that belong to the near infrared and improved the accuracy in the land cover classification.

Based on the obtained results, it was concluded that the proposed method made the feature selection based on the land cover classes of each scene, while RF made the feature selection based on the redundancy of the bands.

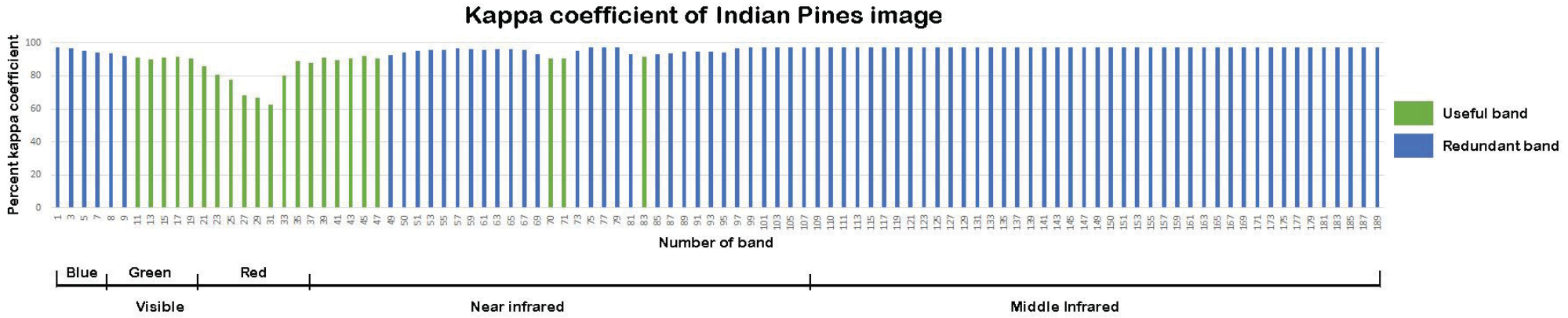
Selecting bands based on the available training samples has advantages, because the selection maximizes the separability between the desired classes. On the other hand, methods that reduce the number of bands based on the analysis of redundancy may produce worse results when these bands are used in the classification. Therefore, better accuracy is expected when the selection is performed with the help of training samples, as it is the case of the SAE-1DCNN model.

GRAPHIC 1 - THRESHOLD DEFINED OF KAPPA COEFFICIENT FOR THE SALINAS BANDS



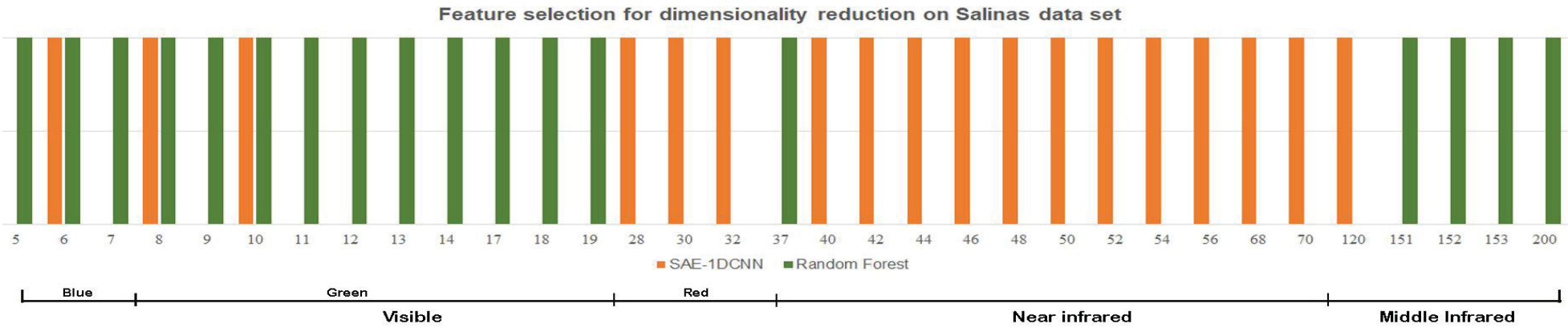
FONT: The author (2021).

GRAPHIC 2 - THRESHOLD DEFINED OF KAPPA COEFFICIENT FOR THE INDIAN PINES BANDS



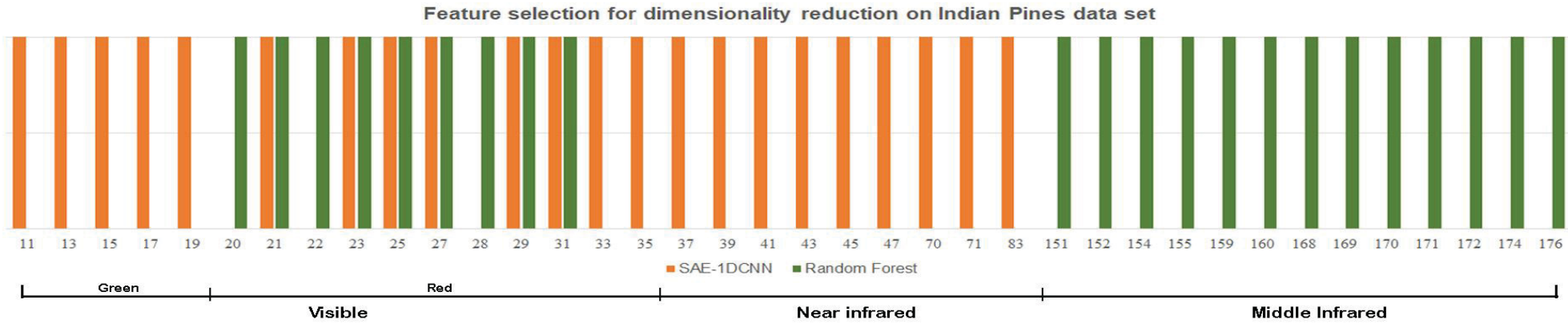
FONT: The author (2021).

GRAPHIC 3 - BAND SELECTION COMPARISON BETWEEN THE RANDOM FOREST AND SAE-1DCNN MODELS FOR THE SALINAS DATA SET



FONT: The author (2021).

GRAPHIC 4 - BAND SELECTION COMPARISON BETWEEN THE RANDOM FOREST AND SAE-1DCNN MODELS FOR THE INDIAN PINES DATA SET



FONT: The author (2021).

4.2 HYPERSPECTRAL DATA PROCESSING

After the pre-processing steps, the remaining bands were used as input for the SAE-1DCNN classifier. The results obtained are presented in the following sub-sections. To evaluate the performance of the method, other established machine learning methods were applied under the same conditions for comparison purposes.

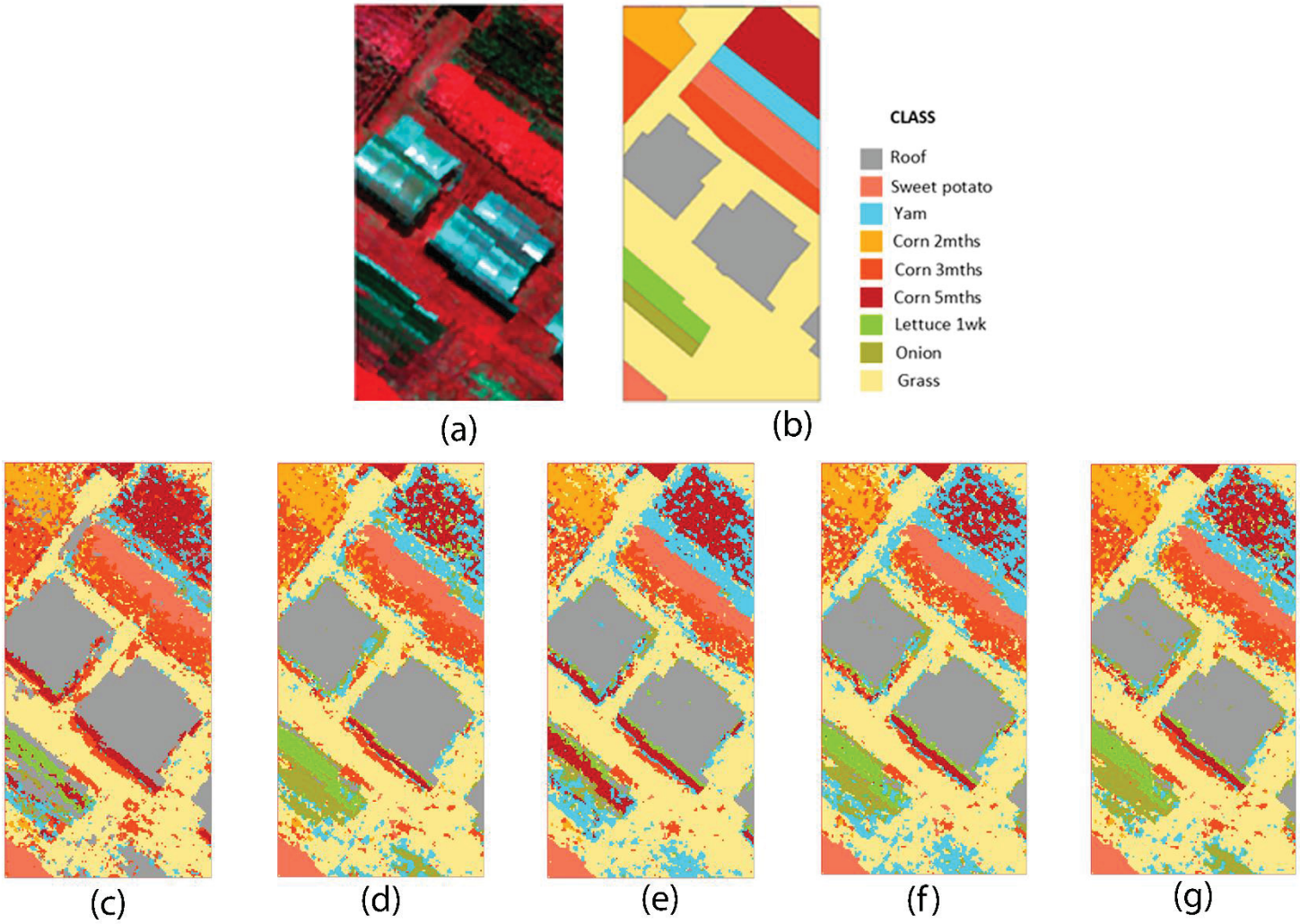
The discussion of the results is divided into two steps. First, the overall accuracy measurements achieved with the five classification approaches are compared and discussed. For this purpose, quality indexes like Overall Accuracy (OA), Average Accuracy (AA) and Kappa Coefficient were computed from the confusion matrix of the verification samples. Then, in a second step, confusion between classes is described in detail analyzing the user's and producer's accuracy obtained in each experiment, which allows a better view of the capacity of each method to deal with spectrally similar classes.

4.2.1 Canguiri Experiment

Figure 22 displays the Cubert's FireflEYE S185 hyperspectral image obtained at the Canguiri Farm, the ground truth map and the classification results obtained by the different machine learning approaches. These thematic maps are related to the statistics listed in Table 17, where the producer's and user's accuracy are compared. This is a more complex scene because there are more similar classes in spectral terms. For example, there are corn fields with different growing seasons (2, 3 and 5 months), as well as other crops. The easiest case is the classification of the roofs because their spectral signature is different from the remaining classes.

The result obtained using SVM is displayed in Figure 22c. The method enabled separating roofs and grass, but had difficulties in classifying the crops. The greatest confusion occurred between the Corn 2mths and Corn 3mths. This is also visible in the accuracy values listed in Table 17. According to Table 17, the Corn 2mths class presented the worst producer's accuracy with a value of 77.27% and the worst user's accuracy was the Corn 3mths with 73.95%. While the Roof class was the best classified with values of 100.00% and 98.35% for producer's and user's accuracy, respectively.

FIGURE 22 - (A) HYPERSPECTRAL IMAGE OF THE CANGUIRI FARM DATA SET (R = 90, G = 60, B = 40). (B) GROUND TRUTH CLASSIFICATION MAP OF CANGUIRI FARM DATA SET. CLASSIFICATION MAPS OBTAINED BY TESTS FOR DIFFERENT MACHINE LEARNING APPROACH OVER CANGUIRI FARM DATA SET. (C) SVM; (D) ANN; (E) CNN; (G) SAE.



FONT: The author (2021).

TABLE 17 - CLASSIFICATION ACCURACY BASED ON THE SUMMARY OF CONFUSION MATRIX FROM PROPOSED METHODS OVER THE CANGUIRI FARM DATA SET

Classification Method	**SVM**		**ANN**		**CNN**		**SAE-1DCNN**		**SAE**	
Figure 22	**(c)**		**(d)**		**(e)**		**(f)**		**(g)**	
Accuracy (%)	**PA**	**UA**	**PA**	**UA**	**PA**	**UA**	**PA**	**UA**	**PA**	**UA**
Roofs	100.00	**98.35**	100.00	88.81	100.00	77.78	100.00	88.81	100.00	89.47
Sweet potato	97.18	**98.57**	97.18	**98.57**	97.18	**98.57**	97.18	**98.57**	97.18	97.18
Yam	78.89	87.65	86.67	88.64	86.67	83.87	88.89	87.91	85.56	**90.59**
Corn 2mths	77.27	**95.77**	85.23	89.29	73.86	92.86	79.55	93.33	86.36	92.68
Corn 3mths	88.89	73.95	87.88	86.14	79.80	85.87	88.89	87.13	85.86	**87.63**
Corn 5mths	92.04	94.55	91.15	**96.26**	88.50	93.46	91.15	94.50	92.04	94.55
Lettuce 1wk	96.08	**96.08**	96.08	94.23	98.04	89.29	94.12	94.12	96.08	**96.08**
Onion	94.74	93.10	98.25	94.92	92.98	96.36	98.25	**98.25**	98.25	96.55
Grass	89.71	85.31	88.97	93.80	87.50	92.97	94.12	94.12	94.85	**94.16**

FONT: The author (2021).
PA= Producer's Accuracy, UA= User's Accuracy

The second thematic map, produced with ANN (Figure 22d), has similar problems. There was confusion between Corn 2mths and Corn 3mths. The results were better when analyzing the accuracy values relating to the previous model. The producer's accuracy of the Corn 2mths class (85.23%) is the lowest, while Corn 3mths had the lowest value in user's accuracy with 86.14%, according to Table 17.

Figure 22e displays the thematic map obtained with the CNN method. The quality of the CNN classifier was worse than the one achieved with ANN; however, it was better than SVM. The confusion between Corn 2mths and Corn 3mths is still visible, as in the previous methods, and between Corn 3mths and Roofs. The producer's accuracy of the Corn 2mths class (73.86%) is the lowest, when considering all the classes, and when comparing all methods. According to Table 17, CNN enabled a particularly good classification of Roofs, with 100.00% producer's accuracy, but on the other hand, the user's accuracy of this class is the lowest (77.78%). This means that this class was overestimated.

The thematic map presented in Figure 22f displays the results obtained with the SAE-1DCNN approach. The confusion between Corn 2mths and Corn 3mths prevailed as in the three previous methods. When analyzing the accuracies obtained for each class, Corn 2mths has the lowest values in producer's accuracy with 79.55%, while the lowest value for user's accuracy were the Yam and Corn 3mths classes with 88.89%. The best producer's accuracy was obtained, again, for the

Roofs class (100.00%). This is expected, as this class is very uniform and spectrally different from the others, which are vegetation classes.

Finally, Figure 22g shows the thematic map obtained with the SAE method. In this method there was confusion between the Yam and Grass classes and also between the Corn 2mths and Corn 3mths classes (the same one that was presented in previous methods). The Yam, Corn 2mths and Corn 3mths classes presented the lowest values in producer's accuracy with 85.56%, 86.36% and 85.86%, respectively, while the highest value was presented again in the Roofs class with 100.00%. The highest value in user's accuracy was presented in the Sweet potato class with 97.18%, while the lowest was for the Corn 3mths class with 87.63%.

As the reliability of a thematic map is described by the user's accuracy that informs how often the class plotted on the map represents the right information on the ground, the user's accuracy was analyzed in the last part of the comparison. It was noted that SVM had the highest user's accuracy for "Roofs" and "Corn 2mths". In addition, this method produced relative worse results for the "Corn 3mths " class when compared to all other methods. Nevertheless, it is worth stating that the user's accuracy achieved for "Roofs" is remarkably superior. ANN produced better results only for "Corn 5mths", but the other methods enable similar results. CNN did not good perform in processing this image, thus, it was not superior in any class. SAE-1DCNN was superior for "Onion", while SAE was superior for "Yam". Both methods are equivalent, and the differences are small as in "Corn 3mths" and "Grass". This indicates that the methods that relate SAE allowed a better differentiation of classes with similar spectral signatures in this scene.

4.2.2 Pavia Experiment

The University of Pavia data set captured by the ROSIS sensor, the ground truth, and the results of the thematic maps of the second experiment are presented in Figure 23. In addition, Table 18 shows the statistical data based on the confusion matrix of the methods developed in this experiment. The Pavia data set covers an urban area with classes with quite different spectral responses, which allows greater separability between classes. This is reflected in the high statistical values listed in Table 18.

The thematic classification of the SVM method can be seen in Figure 23c. Here, there is only one vegetation class: Trees, that can be separated from the remaining classes that are artificial surfaces. When the classes have a very particular signature, such as Painted metal sheets and Shadows, the accuracies are high, which was expected. Spectral confusion can be expected between classes like “Gravel”, “Bitumen”, “Meadows” and “Bricks”. For example, gravel is a mixture of different stones that results in a complex spectral signature that produces spectral confusion with other classes, like “Bricks”. This was a problem for SVM. The ANN method also found some difficulty in dealing with such spectral similarity, which led to confusion again between “Bricks” and “Gravel”. Additionally, there was confusion between the “Bare soil” and “Meadows” and “Bitumen” and “Asphalt” classes in the classified image displayed in Figure 23d.

TABLE 18 - CLASSIFICATION ACCURACY BASED ON THE SUMMARY OF CONFUSION MATRIX FROM PROPOSED METHODS OVER THE UNIVERSITY OF PAVIA DATA SET.

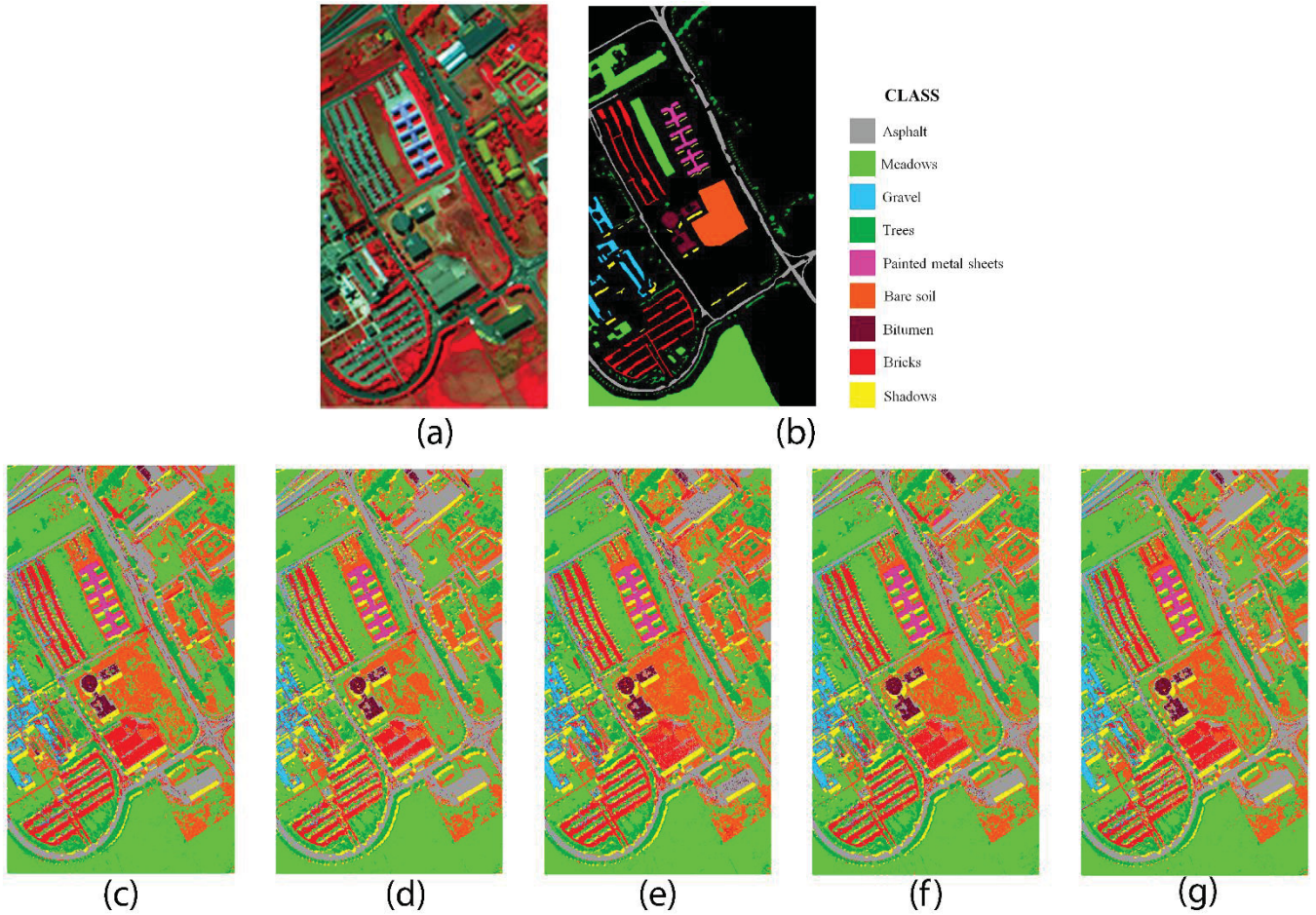
Classification Method	**SVM**		**ANN**		**CNN**		**SAE-1DCNN**		**SAE**	
Figure 23	**(c)**		**(d)**		**(e)**		**(f)**		**(g)**	
Accuracy (%)	**PA**	**UA**	**PA**	**UA**	**PA**	**UA**	**PA**	**UA**	**PA**	**UA**
Asphalt	91.58	94.45	93.73	89.75	92.08	**95.95**	93.33	94.98	94.57	91.21
Meadows	98.35	95.06	98.46	91.47	98.46	**95.83**	98.21	95.38	98.03	93.39
Gravel	80.60	83.36	70.03	86.05	75.39	**87.71**	76.50	87.39	79.50	79.87
Trees	93.33	96.50	90.93	**97.77**	93.88	97.06	94.97	95.92	89.40	97.03
Painted metal sheets	99.74	99.48	99.48	98.96	99.74	99.74	99.74	99.74	99.48	**100.00**
Bare soil	85.21	93.78	71.93	92.63	87.84	**94.48**	85.60	94.21	81.33	92.11
Bitumen	82.05	78.43	83.85	83.21	86.67	85.14	87.69	84.24	82.31	**88.43**
Bricks	90.11	87.11	85.42	85.42	93.82	83.10	91.52	85.48	82.60	**88.04**
Shadows	100.00	**100.00**	100.00	99.28	100.00	99.64	100.00	99.64	100.00	99.64

FONT: The author (2021).
PA= Producer’s Accuracy, UA= User’s Accuracy

The CNN and SAE-1DCNN methods are more suitable for dealing with such confusion, as can be seen in Figures 23e and 23f. Their increased performance is also visible in the accuracy indexes.

For this scene, the SAE method is inferior to the previous methods (CNN and SAE-1DCNN) according to Table 18, although it achieves the highest values for the "Bitumen" and "Bricks" classes in relation to user’s accuracy. Figure 18g shows the thematic map for the SAE method.

FIGURE 23 - (A) HYPERSPECTRAL IMAGE OF THE UNIVERSITY OF PAVIA DATA SET (R = 90, G = 60, B = 40). (B) GROUND TRUTH CLASSIFICATION MAP OF UNIVERSITY OF PAVIA DATA SET. CLASSIFICATION MAPS OBTAINED BY TESTS FOR DIFFERENT MACHINE LEARNING APPROACH OVER UNIVERSITY OF PAVIA DATA SET. (C) SVM; D) ANN; (E) CNN; (F) PROPOSED SAE-1DCNN; (G) SAE



FONT: The Author (2021).

4.2.3 Indian Pines Experiment

The most complex scene, in terms of spectral confusion, is the Indian Pine data set. This was visible in the comparison of the global accuracy indexes. Here the details of the user's and producer's accuracies are observed (Table 19). Figure 24 shows the hyperspectral image of Indian Pines, the ground truth, recorded with 7 land cover classes, and the result of the thematic classifications obtained through the algorithms applied for this study.

According to Figure 24c, the SVM method encountered difficulties in separating soybeans and corn crops. The main confusion was between the Corn-min and Soybeans-min classes. Analyzing the statistics, the worst classified according to the producer's accuracy was Corn-min with 22.52%, being the worst value when compared with all methods used in this study. On the other hand, SVM was very efficient in classifying "Grass" and "Vegetation" pixels.

The thematic map obtained with the ANN method, Figure 24d, shows that the "Corn-min" pixels could not be correctly identified and were classified as "Soybeans-min" at the bottom left of the image. There was also confusion between "Soybeans-min" and "Vegetation", "Corn-min" and "Soybeans-no till" crops that are difficult to separate, which is visible in the low values of these classes in Table 19. Performance in relation to "Vegetation" is lower, when compared to the previous method, which was not expected. On the other hand, as in the previous method, the best results are related to Grass/trees with 99.06% for producer's accuracy and 97.22% for user's accuracy.

Figure 24e displays the result of classification with the CNN method. Here, the biggest confusion occurred between "Corn-min" and "Soybeans-min" has the worst producer's accuracy, while the Grass/trees class has the highest one, with percentages of 99.06%. Consequently, the best labeled class in this thematic map was Grass/trees and can be identified as red regions in Figure 24e.

The confusion between "Corn-min" and "Soybeans-min" is also a problem when applying the proposed method (SAE-1DCNN), as shown in Figure 24f. There were also problems between the classes of "Corn-no till" and "Soybeans-no till" at the time of being classified. Table 19 also shows that the highest value for producer's and user's accuracy was given for the Grass/trees class with 98.11% and 99.05%, respectively. This is the reason why, in the thematic map obtained by this method,

the areas referring to the Grass/trees class are the most uniform and there is no confusion with the other classes.

The thematic map presented in Figure 24g represents the classification obtained by the SAE method. The confusion in this method occurred between the classes "Corn-min" and "Soybeans-min" as it happened in the previous methods. The highest value in relation to user's accuracy was for the class "Grass / trees" with 99.05%, while the lowest was given in the class "Vegetation" with 62.31%. With these results, it was confirmed that the "Grass/ trees" class was the best classified by this method.

TABLE 19 - CLASSIFICATION ACCURACY BASED ON THE SUMMARY OF CONFUSION MATRIX FROM PROPOSED METHODS OVER THE INDIAN PINES DATA SET

Classification Method	**SVM**		**ANN**		**CNN**		**SAE-1DCNN**		**SAE**	
Figure 24	**(c)**		**(d)**		**(e)**		**(f)**		**(g)**	
Accuracy (%)	**PA**	**UA**	**PA**	**UA**	**PA**	**UA**	**PA**	**UA**	**PA**	**UA**
Vegetation	80.90	**98.63**	83.15	47.13	73.03	73.03	83.15	60.66	91.01	62.31
Corn-no till	80.31	77.27	77.17	85.96	71.65	77.78	76.38	**87.39**	83.46	85.48
Corn-min	22.52	71.43	31.53	**89.74**	42.34	57.32	49.55	78.57	47.75	82.81
Grass/Pasture	97.87	76.67	97.87	**77.97**	97.87	68.66	91.49	75.44	95.74	86.54
Grass/trees	100.00	97.25	**99.06**	97.22	99.06	97.22	98.11	99.05	98.11	99.05
Soybeans-no till	75.96	58.09	79.81	63.85	56.73	51.30	83.65	73.73	71.15	**74.00**
Soybeans-min	83.68	69.43	69.47	79.04	72.63	70.41	83.16	**82.72**	82.11	78.39

FONT: The author (2021).

PA= Producer's Accuracy, UA= User's Accuracy

Considering the user's accuracy of each class, ANN produced results that are below the other methods. The best classification of "Vegetation" was obtained using the SVM classifier. When it comes to differentiating Corn and Soybeans, the SAE and SAE-1DCNN methods are better, although the Soybeans-no till class was the most difficult to be discriminated against. Comparing these two methods, one can see that SAE and SAE-1DCNN achieved better user's accuracy for "Grass/trees" with 99.05%, however, SAE achieved better user's accuracy for "Grass / Pasture". The proposed method proved to be more efficient for classifying the "Corn-no till" and "Soybeans-min", while SAE was more efficient for Soybeans-no till classes, although the values computed for these two methods were close.

FIGURE 24 - (A) HYPERSPECTRAL IMAGE OF THE INDIAN PINES DATA SET (R = 50, G = 27, B = 17). (B) GROUND TRUTH CLASSIFICATION MAP OF THE INDIAN PINES DATA SET. CLASSIFICATION MAPS OBTAINED BY TESTS FOR DIFFERENT MACHINE LEARNING APPROACH OVER THE INDIAN PINES DATA SET. (C) SVM; (D) ANN; (E) CNN; (F) PROPOSED SAE-1DCNN; (G) SAE

FONT: The author (2021).

4.3 PERFORMANCE ASSESSMENT

4.3.1 Overall accuracy Comparison

Table 20 displays the quality indexes (Overall Accuracy and Kappa Coefficient) computed from the verification samples of each experiment and the resulting thematic maps are presented in Figures 22-24. Table 20 shows the results for the Canguiri, University of Pavia and Indian Pines images because these were used for the classification of land cover using visual inspection to eliminate noisy bands and RF for dimensionality reduction.

The Pavia scene is, in spectral terms, the simplest image because the spectral signature of the classes is different, which explains the relative high values of the accuracy indexes. The worst results were obtained using ANNs in the Pavia scene, while the best ones were achieved using the CNN method. However, the quality achieved by the SAE-1DCNN method is almost equal, less than 1% difference, which shows that the proposed method is superior to the ANN, SVM, SAE methods and comparable to CNN when it comes to classifying hyperspectral images.

TABLE 20 - COMPARISON OF OVERALL ACCURACY, AVERAGE ACCURACY AND KAPPA COEFFICIENT

Hyperspectral Data set	Index	SVM	ANN	CNN	SAE-1DCNN	SAE
Canguiri Farm	OA	90.29	91.87	88.83	92.35	92.72
	Kappa	88.95	90.75	87.29	**91.30**	**91.71**
University of Paiva	OA	93.35	91.10	94.04	93.82	92.22
	Kappa	91.15	88.04	**92.07**	**91.78**	89.60
Indian Pines	OA	76.10	74.03	71.19	79.84	79.97
	Kappa	71.36	69.30	65.71	**76.04**	**76.14**

FONT: The author (2021).

The more complex data set is the Indian Pines hyperspectral image because the crops are mixed in the fields, which makes it difficult to separate them. Different mixtures result in similar spectral signatures that do not allow one class to be separated from another. This difficulty is reflected in the accuracy values, which are low compared to the other data sets. The lowest values were obtained using CNN,

which is also visible in the thematic maps. On the other hand, the highest accuracies are related to the SAE method. Again, the proposed method (SAE-1DCNN) produces results that are equivalent to those achieved with SAE.

The accuracy indexes obtained applying to the Canguiri data set point out that the CNN approach had more difficulty in classifying the image. This is also visible in the thematic maps presented in Figure 20, where the crops were not correctly classified. The SAE and SAE-1DCNN methods obtained the best results with very similar values, i.e., the OA values obtained were 92.72% and 92.35%, and the Kappa coefficient values were around 91.71% and 91.30%, respectively. Analyzing these statistical results, it can be stated that the SAE-1DCNN method can be applied in hyperspectral images for land cover classification purposes.

4.3.2 Processing time

Based on the previous discussion, it was seen that the methods found difficulties separating some spectrally similar classes, depending on the scene and data set. The best results were obtained using the SAE, but the SAE-1CNN produced similar results in terms of accuracy. To evaluate the advantage of the SAE-1DCNN method in terms of computational efficiency, the processing time was also compared. Table 21 presents the time in seconds (s) used for processing each hyperspectral image.

TABLE 21 - COMPARISON OF PROCESSING TIME BY EACH MACHINE LEARNING METHOD PER HYPERSPECTRAL DATA SET.

Hyperspectral Data set	SVM (s)	ANN (s)	CNN (s)	SAE-1DCNN (s)	SAE (s)
Canguiri Farm	0.21	121.94	245.29	188.95	248.29
University of Paiva	21.39	379.46	6340.46	3811.23	1499.44
Indian Pines	17.69	1711.95	3209.96	2500.77	4892.27

FONT: The author (2021).

SVM and ANN are faster methods but, as verified before, their results are worse. So, they can solve the problem faster, but the accuracy is lower. The SAE and SAE-1DCNN methods demand more processing time but produce better results.

A direct comparison of these methods reveals that the SAE-1DCNN needs less time, including complex and less complex scenes, while the SAE method demands more processing time, although the land cover classification accuracy was higher (minimal difference between the proposed method) for the three scenes evaluated.

Although greater accuracy was obtained with the SAE model (about 0.13% with respect to the SAE-1DCNN) in the Indian Pines experiment, it did not perform well in terms of processing time, demanding the highest processing time. In this context, the efficiency of the SAE-1DCNN model is confirmed, since, in complex scenes with very similar vegetation classes in spectral signatures, for example, Corn-no till and Corn-min; Soybeans-no till and Soybeans-min it managed to discriminate and higher classification accuracy was obtained in user's accuracy for these classes. In addition, it reduced processing time to 40% when comparing these two methods.

5 CONCLUSIONS AND RECOMMENDATIONS

In this research, it was proposed a deep learning method that integrates stacked autoencoders (SAE) and convolutional neural networks (CNN) to classify remote sensing hyperspectral images, the so-called SAE-1DCNN model. The efficiency of the proposed approach was evaluated different aspects: quality of the produced thematic map, processing time, its capacity to detect noisy bands and the possibility to use it as band selection method. The proposed method includes convolutional layers in the encoding and decoding phases of the stacked-autoencoder.

The proposed method was evaluated with different hyperspectral data sets and compared to other classification methods. It was verified that the SAE-1DCNN model enables higher land cover classification accuracy when compared to the other machine learning models used in this study. The proposed method, called SAE-1DCNN, can be considered a valid alternative for classification of hyperspectral images considering different types of scenes such as urban, crop fields and mixed agriculture. These results were evaluated under two criteria. The first one was the accuracy of the land cover classification, which proved that the method is effective and efficient. The second criterion that was analyzed was processing time.

In terms of accuracy, the best results were obtained with traditional autoencoders, but the proposed method produced similar results, with less processing time. Although the results obtained relating the classification accuracy by the SAE model were higher compared to the SAE-1DCNN model (minimum difference) for the three hyperspectral images evaluated in this study, the processing time needed by the SAE-1DCNN model is lower when compared to the SAE model, confirming that the proposed model is an alternative for the hyperspectral images classification. Although the results obtained proved its potential for hyperspectral image classification, it should be noted that they depend highly on the samples (training and testing), which must be chosen with care.

It was also proved that the proposed method can detect noisy bands and cancel them in the classification step. This is useful because it can be used to avoid human band selection, based on visual analysis of each spectral band. Thus, this method

can be considered as a scientific alternative for the identification of noisy bands without relying on visual identification.

The experiments also showed that the method can be used to perform band selection based on the training samples. This was possible because the net computes features based on the spectral signature of the training pixels and then selects the more relevant features in the deep net. It is concluded that this method reduces the spectral dimension based on the reflectance of the evaluated classes of the scene and not based on band redundancy. This dimensionality reduction capacity, based on the training samples, allows increasing the accuracy in the image classification.

It is recommended that the method is evaluated with different images, including other classes, and with different spectral and spatial resolutions. In future studies, it is recommended to assess if the band selection method is also successful when other classification methods are applied in the classification phase. In this research, band selection and noise detection were performed with an empirical threshold. Future studies should be devoted to propose a method to set this threshold according to the input variables and information.

Finally, it is recommended that new studies be devoted to the architecture of the net, varying the number of layers and neurons, which was not possible in the present study.

ACKNOWLEDGEMENTS

The author would like to thank to Karlsruhe Institute of Technology (KIT) and the Institute of Photogrammetry and Remote Sensing (IPF) in Karlsruhe-Germany for making available the UAV (Unmanned Aerial Vehicle) equipment for obtaining the Cubert's FireflEYE S185 hyperspectral data set in the Canguiri Farm of the Federal University of Paraná (UFPR) in Paraná-Brazil; Purdue University, USA, for providing the AVIRIS hyperspectral data, and Prof. P. Gamba for providing the ROSIS data from Pavia, Italy. Furthermore, CNPq/Brazil for the financial support. In addition, the author would also like to thank to CNPq - Brazil for the financial support during the research and Karlsruhe House of Young Scientists (KHYS) and the German Academic Exchange Service (DAAD) for the financing during the internship carried out at KIT - IPF in Germany.

REFERENCES

AHMED, B.; KAMRUZZAMAN, M.; ZHU, X.; RAHMAN, M.; CHOI, K. **Simulating land cover changes and their impacts on land surface temperature in Dhaka, Bangladesh**. Remote Sensing, 5(11), 5969-5998, 2013.

ATKINSON, P. M.; TATNALL, A. R. **Introduction neural networks in remote sensing.** International Journal of remote sensing, 18(4), 699-709, 1997

BALL, J. E.; ANDERSON, D. T.; CHAN, C. S. **Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community.** Journal of Applied Remote Sensing, 11(4), 042609, 2017.

BASAEED, E.; BHASKAR, H.; HILL, P.; AL-MUALLA, M.; BULL, D. **A supervised hierarchical segmentation of remote-sensing images using a committee of multi-scale convolutional neural networks**. International Journal of Remote Sensing, 37(7), 1671-1691, 2016.

BELGIU, M.; DRĂGUŢ, L. **Random Forest in remote sensing: A review of applications and future directions.** ISPRS Journal of Photogrammetry and Remote Sensing, 114, 24–31, 2016.

BENEDIKTSSON, J. A.; GHAMISI P. Spectral-Spatial Classification of Hyperspectral Remote Sensing Images, First Edition, Artech House, 258p, 2015.

BENGIO, Y. **Deep learning of representations for unsupervised and transfer learning**. In Proceedings of ICML workshop on unsupervised and transfer learning (pp. 17-36), 2012.

BREIMAN, L. **Random forests**. Machine learning, 45(1), 5-32, 2001.

CHEN, X.; XIANG, S.; LIU, C. L.; PAN, C. H. **Vehicle detection in satellite images by hybrid deep convolutional neural networks**. IEEE Geoscience and remote sensing letters, 11(10), 1797-1801, 2014.

CHEN, Y.; JIANG, H.; LI, C.; JIA, X.; GHAMISI, P. **Deep feature extraction and classification of hyperspectral images based on convolutional neural networks**. IEEE Transactions on Geoscience and Remote Sensing, 54(10), 6232-6251, 2016.

CHEN, Y.; LIN, Z.; ZHAO, X.; WANG, G.; GU, Y. **Deep learning-based classification of hyperspectral data.** IEEE Journal of Selected topics in applied earth observations and remote sensing, 7(6), 2094-2107, 2014.

CHEN, Y.; ZHAO, X.; JIA, X. **Spectral–spatial classification of hyperspectral data based on deep belief network**. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 8(6), 2381-2392, 2015.

CHENG, G.; HAN, J.; LU, X. **Remote sensing image scene classification: Benchmark and state of the art.** Proceedings of the IEEE, 105(10), 1865-1883, 2017.

DENG, L. **A tutorial survey of architectures, algorithms, and applications for deep learning**. APSIPA Transactions on Signal and Information Processing, 3, 2014.

DIAO, W.; SUN, X.; DOU, F.; YAN, M.; WANG, H.; FU, K. **Object recognition in remote sensing images using sparse deep belief networks.** Remote Sensing Letters, 6(10), 745-754, 2015.

DONAHUE, J.; JIA, Y.; VINYALS, O.; HOFFMAN, J.; ZHANG, N.; TZENG, E.; DARRELL, T. **Decaf: A deep convolutional activation feature for generic visual recognition**. In International conference on machine learning (pp. 647-655), 2014.

DU, B.; XIONG, W.; WU, J.; ZHANG, L.; ZHANG, L.; TAO, D. **Stacked convolutional denoising auto-encoders for feature representation**. IEEE transactions on cybernetics, 47(4), 1017-1027, 2016.

FENG, Q.; LIU, J.; GONG, J. **UAV remote sensing for urban vegetation mapping using random forest and texture analysis.** Remote sensing, 7(1), 1074-1094, 2015.

FENG, Q.; ZHU, D.; YANG, J.; LI, B. **Multisource hyperspectral and lidar data fusion for urban land-use mapping based on a modified two-branch convolutional neural network**. ISPRS International Journal of Geo-Information, 8(1), 28, 2019.

GARCIA-SALGADO, B. P.; PONOMARYOV, V. I.; SADOVNYCHIY, S.; ROBLES-GONZALEZ, M. **Parallel supervised land-cover classification system for hyperspectral and multispectral images**. Journal of Real-Time Image Processing, 15(3), 687-704, 2018.

GENG, J.; FAN, J.; WANG, H.; MA, X.; LI, B.; CHEN, F. **High-resolution SAR image classification via deep convolutional autoencoders**. IEEE Geoscience and Remote Sensing Letters, 12(11), 2351-2355, 2015.

GISLASON, P. O.; BENEDIKTSSON, J. A.; SVEINSSON, J. R. **Random forests for land cover classification**. Pattern Recognition Letters, 27(4), 294-300, 2006.

GONG, B.; IM, J.; MOUNTRAKIS, G. **An artificial immune network approach to multi-sensor land use/land cover classification**. Remote Sensing of Environment, 115(2), 600-614, 2011.

GONG, M.; YANG, H.; ZHANG, P. **Feature learning and change feature classification based on deep learning for ternary change detection in SAR images**. ISPRS Journal of Photogrammetry and Remote Sensing, 129, 212-225, 2017.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep learning**. First Edition, The MIT press, Cambridge, Massachusetts, 801p, ISBN 9780262035613, 2016.

HARA, K.; SAITO, D.; SHOUNO, H. **Analysis of Function of Rectified Linear Unit Used in Deep learning**. in International Joint Conference on Neural Networks (IJCNN). IEEE: Killarney, IRELAND. 144–151, 2015.

HASSAN, M. M. 2017. **Monitoring land use/land cover change, urban growth dynamics and landscape pattern analysis in five fastest urbanized cities in**

Bangladesh. Remote Sensing Applications: Society and Environment, 7, 69-83, 2017.

HAYKIN, S. **Neural networks and learning machines**, Third Edition, Pearson Prentice Hall Education, Upper Saddle River, New Jersey 07458, 937p, ISBN-13: 978-0-13-147139-9, 2008.

HENITS, L.; JÜRGENS, C.; MUCSI, L. **Seasonal multitemporal land-cover classification and change detection analysis of Bochum, Germany, using multitemporal Landsat TM data.** International Journal of Remote Sensing, 37(15), 3439-3454, 2016.

HOSSAIN, M.; REKABDAR, B.; LOUIS, S. J.; DASCALU, S. **Forecasting the weather of Nevada: A deep learning approach**. In 2015 international joint conference on neural networks (IJCNN), IEEE, (pp. 1-6), 2015.

HU, W.; HUANG, Y.; WEI, L.; ZHANG, F.; LI, H. **Deep convolutional neural networks for hyperspectral image classification**. Journal of Sensors, 2015.

HUANG, B.; ZHAO, B.; SONG, Y. **Urban land-use mapping using a deep convolutional neural network with high spatial resolution multispectral remote sensing imagery**. Remote Sensing of Environment, 214, 73-86, 2018.

HUGHES, G.F**. On the mean accuracy of statistical pattern recognizers**. IEEE Trans. Inform. Theory IT 14, 55–63, 1968.

ISIK, S.; KALIN, L.; SCHOONOVER, J. E.; SRIVASTAVA, P.; LOCKABY, B. G. **Modeling effects of changing land use/cover on daily streamflow: an artificial neural network and curve number based hybrid approach**. Journal of Hydrology, 485, 103-112, 2013.

JIA, K.; LIANG, S.; LIU, S.; LI, Y.; XIAO, Z.; YAO, Y.; JIANG, B.; ZHAO, X.; WANG, X.; XU, S.; CUI, J. **Global land surface fractional vegetation cover estimation using general regression neural networks from MODIS surface reflectance**. IEEE Transactions on Geoscience and Remote Sensing, 53(9), 4787-4796, 2015.

JING, R.; LIU, S.; GONG, Z.; WANG, Z.; GUAN, H.; GAUTAM, A.; ZHAO, W. **Object-based change detection for VHR remote sensing images based on a Trisiamese-LSTM**. International Journal of Remote Sensing, 41(16), 6209-6231, 2020.

KEMKER, R.; SALVAGGIO, C.; KANAN, C. **Algorithms for semantic segmentation of multispectral remote sensing imagery using deep learning**. ISPRS journal of photogrammetry and remote sensing, 145, 60-77, 2018.

KHAN, M.; JAN, B.; FARMAN, H. **Deep Learning: Convergence to Big Data Analytics**. First Edition, Springer Singapore, 93p, 2019.

KIM, Y.; MOON, T. Human detection and activity classification based on micro-Doppler signatures using deep convolutional neural networks. IEEE geoscience and remote sensing letters, 13(1), 8-12, 2015.

KIRANYAZ, S.; AVCI, O.; ABDELJABER, O.; INCE, T.; GABBOUJ, M.; INMAN, D. **1D convolutional neural networks and applications: A survey**. Mechanical Systems and Signal Processing, 151, 107398, 2020.

KUSSUL, N.; LAVRENIUK, M.; SKAKUN, S.; SHELESTOV, A. **Deep learning classification of land cover and crop types using remote sensing data.** IEEE Geoscience and Remote Sensing Letters, 14(5), 778-782, 2017.

LANDIS, J. R.; KOCH, G. G. **The measurement of observer agreement for categorical data**. Biometrics, 159-174, 1977.

LECUN, Y.; BENGIO, Y.; HINTON, G. **Deep learning.** Nature, 521(7553), 436, 2015.

LI, W.; DU, Q.; ZHANG, B. **Combined sparse and collaborative representation for hyperspectral target detection**. Pattern Recognition, 48(12), 3904-3916, 2015.

LI, W.; WU, G.; ZHANG, F.; DU, Q. **Hyperspectral image classification using deep pixel-pair features**. IEEE Transactions on Geoscience and Remote Sensing, 55(2), 844-853, 2016.

LI, J.; ZHAO, X.; LI, Y.; DU, Q.; XI, B.; HU, J. **Classification of hyperspectral imagery using a new fully convolutional neural network**. IEEE Geoscience and Remote Sensing Letters, 15(2), 292-296, 2018.

LIN, S. Y.; CHIANG, C. C.; LI, J. B.; HUNG, Z. S.; CHAO, K. M. **Dynamic fine-tuning stacked auto-encoder neural network for weather forecast**. Future Generation Computer Systems, 89, 446-454, 2018.

MA, L.; LIU, Y.; ZHANG, X.; YE, Y.; YIN, G.; JOHNSON, B. A. **Deep learning in remote sensing applications: A meta-analysis and review**. ISPRS Journal of Photogrammetry and Remote Sensing, 152, 166-177, 2019.

MA, X.; GENG, J.; WANG, H. **Hyperspectral image classification via contextual deep learning**. EURASIP Journal on Image and Video Processing, 2015(1), 20, 2015

MAGGIORI, E.; TARABALKA, Y.; CHARPIAT, G.; ALLIEZ, P. **Convolutional neural networks for large-scale remote-sensing image classification.** IEEE Transactions on Geoscience and Remote Sensing, 55(2), 645-657, 2017.

MAHESH, S.; JAYAS, D. S.; PALIWAL, J.; WHITE, N. D. G. **Hyperspectral imaging to classify and monitor quality of agricultural materials**. Journal of Stored Products Research, 61, 17-26, 2015.

MARKOFF, J. **Scientists see promise in deep-learning programs.** New York Times, 23, 2012.

MCCULLOCH, W. S.; PITTS, W. **A logical calculus of the ideas immanent in nervous activity.** The bulletin of mathematical biophysics, 5(4), 115-133, 1943.

MEI, S.; JI, J.; GENG, Y.; ZHANG, Z.; LI, X.; DU, Q**. Unsupervised Spatial–Spectral Feature Learning by 3D Convolutional Autoencoder for Hyperspectral Classification**. IEEE Transactions on Geoscience and Remote Sensing, 57(9), 6808-6820, 2019.

MERKLE, N.; AUER, S.; MÜLLER, R.; REINARTZ, P. **Exploring the potential of conditional adversarial networks for optical and SAR image matching**. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 11(6), 1811-1820, 2018.

NEGNEVITSKY, M. **Artificial intelligence: a guide to intelligent systems.** Second Edition, Pearson education, 435p, ISBN 0 321 20466 2, 2005.

NEAGOE, V. E.; NEGHINA, M.; DATCU, M. **Neural network techniques for automated land-cover change detection in multispectral satellite time series imagery**. International Journal of Mathematical Models and Methods in Applied Sciences, 6(1), 130-139, 2012.

NOGUEIRA, K.; PENATTI, O. A.; DOS SANTOS, J. A. **Towards better exploiting convolutional neural networks for remote sensing scene classification**. Pattern Recognition, 61, 539-556, 2017.

NWANKPA, C.; IJOMAH, W.; GACHAGAN, A.; MARSHALL, S. **Activation functions: Comparison of trends in practice and research for deep learning**. arXiv preprint arXiv:1811.03378, 2018.

OTHMAN, E.; BAZI, Y.; ALAJLAN, N.; ALHICHRI, H.; MELGANI, F. **Using convolutional features and a sparse autoencoder for land-use scene classification.** International Journal of Remote Sensing, 37(10), 2149-2167, 2016.

PAL, M. **Extreme-learning-machine-based land cover classification**. International Journal of Remote Sensing, 30(14), 3835-3841, 2009.

PANDA, S. S.; AMES, D. P.; PANIGRAHI, S. **Application of vegetation indices for agricultural crop yield prediction using neural network techniques**. Remote Sensing, 2(3), 673-696, 2010.

PAOLETTI, M. E.; HAUT, J. M.; PLAZA, J.; PLAZA, A. **A new deep convolutional neural network for fast hyperspectral image classification**. ISPRS Journal of Photogrammetry and Remote Sensing, 145, 120-147, 2018.

PAOLETTI, M. E.; HAUT, J. M.; PLAZA, J.; PLAZA, A. **Deep learning classifiers for hyperspectral imaging: A review.** ISPRS Journal of Photogrammetry and Remote Sensing, 158, 279-317, 2019.

PASHAEI, A.; GHATEE, M.; SAJEDI, H. **Convolution neural network joint with mixture of extreme learning machines for feature extraction and classification of accident images.** Journal of Real-Time Image Processing, 17(4), 1051-1066, 2020.

REDDY, D. S.; PRASAD, P. R. C. **Prediction of vegetation dynamics using NDVI time series data and LSTM.** Modeling Earth Systems and Environment, 4(1), 409-419, 2018.

RICHARDS, J. A. **Remote sensing digital image analysis An Introduction.** Fifth Edition, Springer, 2013.

SCHMIDHUBER, J. **Deep learning in neural networks: An overview**. Neural networks, 61, 85-117, 2015.

SCHOLKOPF B.; SMOLA A. J. **Learning with kernels: support vector machines, regularization, optimization, and beyond**, MIT Press, 2002.

SERPICO, S. B.; D'INCA, M.; MELGANI, F.; MOSER, G. **Comparison of feature reduction techniques for classification of hyperspectral remote sensing data**. In Proceedings of SPIE. Image and Signal Processing of Remote Sensing VIII, 4885, 347–358, 2003.

SHARMA, A.; LIU, X.; YANG, X.; SHI, D. A patch-based convolutional neural network for remote sensing image classification. Neural Networks, 95, 19-28, 2017

SHARMA, S.; SHARMA, S.; ATHAIYA, A. **Activation functions in neural networks**. International Journal of Engineering Applied Sciences and Technology, Vol. 4, Issue 12, ISSN No. 2455-214, 2020.

SUMBUL, G.; CINBIS, R. G.; AKSOY, S. **Fine-grained object recognition and zero-shot learning in remote sensing imagery**. IEEE Transactions on Geoscience and Remote Sensing, 56(2), 770-779, 2017.

UL HAQ, Q. S.; TAO, L.; YANG, S. **Neural network based adaboosting approach for hyperspectral data classification**. In Proceedings of 2011 International Conference on Computer Science and Network Technology (Vol. 1, pp. 241-245). IEEE, 2011.

VAPNIK, V. N. **Statistical learning theory**. John Wiley / Sins, New York, NY, 1998.

WANG, L.; ZHANG, J.; LIU, P.; CHOO, K. K. R.; HUANG, F. **Spectral–spatial multi-feature-based deep learning for hyperspectral remote sensing image classification**. Soft Computing, 21(1), 213-221, 2017.

WANG, S. H; PHILLIPS, P.; SUI, Y.; LIU, B.; YANG, M.; CHENG, H. **Classification of Alzheimer's disease based on eight-layer convolutional neural network with leaky rectified linear unit and max pooling**. Journal of medical systems, 42(5), 85, 2018.

WANG, H.; ZHAO, X.; ZHANG, X.; WU, D.; DU, X. **Long time series land cover classification in China from 1982 to 2015 based on Bi-LSTM deep learning**. Remote Sensing, 11(14), 1639, 2019

WANG, Y.; YAO, H.; ZHAO, S. **Auto-encoder based dimensionality reduction**. Neurocomputing, 184, 232-242, 2016.

WIMMERS, A.; VELDEN, C.; COSSUTH, J. H. **Using deep learning to estimate tropical cyclone intensity from satellite passive microwave imagery**. Monthly Weather Review, 147(6), 2261-2282, 2019.

WU, H.; LIU, B.; SU, W.; ZHANG, W.; SUN, J. **Deep filter banks for land-use scene classification**. IEEE Geoscience and Remote Sensing Letters, 13(12), 1895-1899, 2016.

XING, C.; MA, L.; YANG, X. **Stacked denoise autoencoder based feature extraction and classification for hyperspectral images**. Journal of Sensors, 2016.

XING, Y.; WANG, M.; YANG, S.; JIAO, L. **Pan-sharpening via deep metric learning.** ISPRS Journal of Photogrammetry and Remote Sensing, 145, 165-183, 2018.

XING, H.; MENG, Y.; WANG, Z.; FAN, K.; HOU, D. **Exploring geo-tagged photos for land cover validation with deep learning**. ISPRS journal of photogrammetry and remote sensing, 141, 237-251, 2018.

XU, F.; CERVONE, G.; FRANCH, G.; SALVADOR, M. **Multiple geometry atmospheric correction for image spectroscopy using deep learning**. Journal of Applied Remote Sensing, 14(2), 024518, 2020.

YUE, J.; MAO, S.; LI, M. **A deep learning framework for hyperspectral image classification using spatial pyramid pooling**. Remote Sensing Letters, 7(9), 875-884, 2016.

ZABALZA, J.; REN, J.; ZHENG, J.; ZHAO, H.; QING, C.; YANG, Z.; MARSHALL, S. **Novel segmented stacked autoencoder for effective dimensionality reduction and feature extraction in hyperspectral imaging**. Neurocomputing, 185, 1-10, 2016.

ZHANG, C.; PAN, X., LI, H.; GARDINER, A.; SARGENT, I.; HARE, J.; ATKINSON, P. M. **A hybrid MLP-CNN classifier for very fine resolution remotely sensed image classification.** ISPRS Journal of Photogrammetry and Remote Sensing, 140, 133-144, 2018.

ZHANG, C.; SARGENT, I.; PAN, X.; LI, H.; GARDINER, A.; HARE, J.; ATKINSON, P. M. **Joint Deep Learning for land cover and land use classification**. Remote sensing of environment, 221, 173-187, 2019.

ZHAO, W.; DU, S. **Learning multiscale and deep representations for classifying remotely sensed imagery.** ISPRS Journal of Photogrammetry and Remote Sensing, 113, 155-165, 2016a.

ZHAO, W.; DU, S. **Spectral–spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach.** IEEE Transactions on Geoscience and Remote Sensing, 54(8), 4544-4554, 2016b.

ZHU, X. X.; TUIA, D.; MOU, L.; XIA, G. S.; ZHANG, L.; XU, F.; FRAUNDORFER, F. **Deep learning in remote sensing: A comprehensive review and list of resources.** IEEE Geoscience and Remote Sensing Magazine, 5(4), 8-36, 2017.

ZHU, Z.; WOODCOCK, C. E. **Continuous change detection and classification of land cover using all available Landsat data**. Remote sensing of Environment, 144, 152-171, 2014.

ZHU, Z.; ZHANG, J.; YANG, Z.; ALJADDANI, A. H.; COHEN, W. B.; QIU, S.; ZHOU, C. **Continuous monitoring of land disturbance based on Landsat time series**. Remote Sensing of Environment, 238, 111116, 2020.