1D-CONVOLUTIONAL AUTOENCODER BASED HYPERSPECTRAL DATA COMPRESSION

Jannick Kuester, Wolfgang Gross, Wolfgang Middelmann

Fraunhofer IOSB, Ettlingen, Fraunhofer Institute of Optronics, System Technologies and Image Exploitation

KEY WORDS: hyperspectral imaging, hyperspectral compression, feature extraction, dimensionality reduction, convolutional autoencoder.

ABSTRACT:

Hyperspectral sensor technology has been advancing in recent years and become more practical to tackle a variety of applications. The arising issues of data transmission and storage can be addressed with the help of compression. To minimize the loss of important information, high spectral correlation between adjacent bands is exploited. In this paper, we introduce an approach to compress hyperspectral data based on a 1D-Convolutional Autoencoder. Compression is achieved through reducing correlation by transforming the spectral signature into a low-dimensional space, while simultaneously preserving the significant features. The focus lies on compression of the spectral dimension. The spatial dimension is not used in the compression in order not to falsify correlation between the spectral dimension and accuracy of the reconstruction. The proposed 1D-Convolutional Autoencoder efficiently finds and extracts features relevant for compression. Additionally, it can be exploited as a feature extractor or for dimensionality reduction. The hyperspectral data sets *Greding Village* and *Pavia University* were used for the training and the evaluation process. The reconstruction accuracy is evaluated using the Signal to Noise Ratio and the Spectral Angle. Additionally, a land cover classification using a multi-class Support Vector Machine is used as a target application. The classification performance of the original and reconstructed data are compared. The reconstruction accuracy of the 1D-Convolutional Autoencoder outperforms the Deep Autoencoder and Nonlinear Principal Component Analysis for the used metrics and for both data sets using a fixed compression ratio.

1. INTRODUCTION

Hyperspectral sensors measures the reflected electromagnetic spectrum of a territory in hundreds of narrow and contiguous wavelength intervals, referred to as bands. The resulting data is represented as a 3D data cube, which is composed of two spatial and one spectral dimension. Each pixel of the hyperspectral image contains a spectral signature of the materials in the scene. The signatures of different materials have characteristic features, which can be used to robustly classify land cover or to increase the accuracy of object recognition. Due to the increasing demand for real-time processing of hyperspectral data and industry-related applications, the enormous amounts of data resulting from the high spectral dimensionality by the multiple spectral channels must be addressed. Efficient data transmission and storage are required, as a single hyperspectral data set can have several hundred megabytes. The data volume makes the downlink from a carrier platform to a ground station more complicated, since the bandwidth is limited and can only be expanded with great effort, especially for satellite-based systems. Transmission of large amounts of data is also essential for real-time applications, such as disaster management. In this work, we address the transmission problem through hyperspectral data compression. We focus on lossy compression to optimize towards higher compression rates, while still maintaining the relevant features. The goal is to maintain all relevant features and properties of the hyperspectral data through the compression and reconstruction process to make the data usable for various applications.

Most state-of-the-art methods for lossy compression are based on transform coding. The idea is to transform the hyperspectral data into a low-dimensional space in which the representation of the data is less correlated. There are methods using the 3D-transformations like 3D-discrete wavelet transform (DWT)(Lim et al., 2001) or 3D-discrete cosine transformation (DCT) (Abousleman et al., 1995); others examine the spectral and spatial dimensions separately. In (Penna et al., 2006), a low-complexity version of the Karhunen-Loève transform is used to decorrelate the spectral information, while the JPEG 2000 is used for the spatial decorrelation as well as an entropy coder. In (Du and Fowler, 2007), a *Principal Component Analysis* (PCA) is used to reduce the spectral dimensionality of the hyperspectral data. Following the PCA, a 2D-DWT is applied to the spatial dimension exploiting the spatial correlations for compression. The coefficients are encoded in a bitstream with the JPEG 2000 framework and transmitted to the decoder. In (Du and Fowler, 2008), several strategies for reducing the computational effort of the PCA algorithm are explored. The first method reduces the computational effort by both spatial and spectral subsampling in the covariance calculation. The second method is based on a simple *neural network* (NN) architecture. The NN has a feedforward structure with one input layer and one output layer. The output neuron provides the largest eigenvalue with the weights representing the corresponding eigenvector.

In recent years, machine learning methods have been established to be a powerful tool in signal processing. In (Theis et al., 2017), lossy image compression is performed using an Autoencoder. Since then, Autoencoder approaches have become increasingly popular in hyperspectral image processing. In (Zabalza et al., 2016), stacked autoencoders are used for dimensionality reduction and feature extraction in hyperspectral imaging. The focus lies on dividing the spectral dimension into different regions, which are used by the autoencoder for feature extraction. The extracted features are used to improve the classification results. In (Li and Liu, 2019), a *convolutional neural*

network (CNN) reduces the spectral dimension of multispectral data. The reduced data is then decomposed into coefficients with the DCT and encoded in a bitstream. The CNN is utilized for dimensionality reduction. The actual compression, however, is realized by the DCT and the coding of the bitstream. In (Priya et al., 2019), fully-connected autoencoder approaches are used for dimensionality reduction in hyperspectral imaging. Here, the spectral dimension is transformed to a lower dimensional subspace while simultaneously preserving the significant features. The reconstruction accuracy is examined depending on the number of nodes in the *bottleneck*. In (Licciardi et al., 2014) and (Licciardi and Chanussot, 2018) a *Nonlinear Principal Component Analysis* (NLPCA) based *multilayer perceptron* with an autoencoder architecture is implemented to compress the spectral dimension of hyperspectral data. The autoencoder with non-linear activation functions in the hidden layers resembles a NLPCA with a fixed number of components. The number of components in the input and low-dimensional space determines the compression rate. Due to the non-linear properties of the NLPCA and its fixed number of components, the information of the original spectrum is obtained and distributed equally among all components. In (Kuester et al., 2020), a Deep Autoencoder is used to compress the spectral dimension by reducing the correlation of the hyperspectral data while maintaining significant features with a focus on minimizing the reconstruction error. The reconstruction accuracy is evaluated using the Signal to Noise Ratio (SNR) and the Spectral Angle Mapper (SAM).

This paper investigates the compression performance and the reconstruction accuracy for the spectral dimension using a *1D-Convolutional Autoencoder* (1D-CAE). The compression is achieved through reducing the correlation by transforming it into a low-dimensional space while simultaneously preserving the significant features. The focus lies on the compression of the spectral dimension without taking spatial neighborhoods into account. The spatial dimension is not used in the compression in order not to falsify the correlation between the spectral dimension and the accuracy of the reconstruction. To find relevant information and retain only the essential features for the compression, the model uses a convolutional layer and a max pooling layer. The accuracy of the reconstructed data from the proposed model is compared to the results from the *Deep Autoencoder* (DAE) (Kuester et al., 2020) and the NLPCA method (Licciardi and Chanussot, 2018). The *Signal to Noise Ratio* (SNR) and the *Spectral Angle Mapper* (SAM) are used to calculate the spectral similarity between the reconstructed data set and the original. A land cover classification is used as a target application. For this we use a multi-class *Support Vector Machine* (SVM). The reconstruction accuracy of the 1D-convolutional Autoencoder outperforms the DAE and NLPCA for every used metric for the *Greding Villiage* as well as *Pavia University* hyperspectral data set. Due to the fact that the test data are unknown to the 1D-CAE, we can conclude that the model provides robust generalization.

This paper is structured as follows: Section 2 describes the fundamentals of the AE, as well as the characteristics and the architecture of the 1D-CAE model. Section 3 describes the hyperspectral data sets used for the evaluation. Additionally, the metrics for comparing the spectral signatures are introduced. The results are presented and discussed in Section 4. Section 5 gives an outlook on possible improvements of the compression model.

2. PROPOSED METHOD

In this section, we explain our proposed 1D-CAE structure and the compression process. The 1D-CAE exploits the high interband correlation in the spectral dimension of the hyperspectral data for compression. Compression is realized by reducing the correlation through a transformation of the input spectral signature to a low-dimensional space, while maintaining the significant features. In the low-dimensional space the features are less correlated. Important features and the associated transformation is automatically learned during the training process by the 1D-CAE model. The samples of the spectral signature, referred to as bands, correspond to the features which are reduced by the compression. For this purpose, the 1D-CAE uses unsupervised learning where the input signal is used as a reference for the evaluation of the reconstructed signal. Thus, the 1D-CAE does not need any labeled data for the training process. The model's architecture consists of an Input layer, twelve hidden layers and an Output layer, as shown in Table 1, and can be divided into two parts. The Encoder part describes the encoder function, which transforms the input signal $\mathbf{x} \in \mathbb{R}^b$ to a low-dimensional representation $\mathbf{h} \in \mathbb{R}^{\hat{b}}$ with $\hat{b} \ll b$. The task of the decoder is to reconstruct the input signal $\mathbf{x} \approx \tilde{\mathbf{x}} \in \mathbb{R}^b$ as precisely as possible from the reduced representation $\mathbf{h} \in \mathbb{R}^{\hat{b}}$. The decoder structure mirrors the encoder structure in reverse order.

The encoder is structured as follows: If the input signal has an odd number of bands b, the zero padding layer follows the input layer. This layer adds a zero to the beginning of the input signal to create an even number of samples. If the number of bands is even, this step is skipped. A combination of a 1D-convolutional layer and a max pooling layer is then added to the encoder structure, as shown in Table 1. The 1D-convolutions are used for recognizing local features from the spectral signature. For this, a filter with a defined kernel size is shifted over the samples of the spectral dimension. The 1D-convolutional layer is only used to find features and not to reduce the features. To gather different features we use multiple filters in parallel. Figure 1 illustrates the convolutional filters in the hidden layer by different colors. Each filter is initialized with random weights, which results in different features. After the 1D-convolutional layer follows a max pooling layer. Pooling is used to only pass on the most relevant features to the next layers. For this purpose, a filter with the kernel size is shifted over the output of the 1D-convolutional layer, and the maximum value that corresponds to the feature found within the filter is transferred to the output. Max pooling is a down-sampling operation on feature maps to reduce dimensionality while maintaining characteristic features. This results in a more abstract and compressed representation of the content; additionally, it reduces the number of parameters in the 1D-CAE. By reducing features, we achieve compression within the 1D-CAE. In the first set of 1D-convolutional layer and max pooling layer the number of filters increased and the feature space is reduced. The last column of Table 1 shows the progression of the number of features and filters throughout the 1D-CAE. The following combination of the 1D-convolutional layer and the max pooling layer reduces both the feature space and the number of filters. The encoder structure is expanded by two additional 1D-convolutional layers, which decreases the number of filters to 1. The feature space is not reduced any further because the previously defined compression factor has already been achieved by the two max pooling layers. The compression is achieved by the fact that the input spectral signature is represented through a lower number of significant features, which is called compressed representation $\mathbf{h} \in \mathbb{R}^{\hat{b}}$, as shown
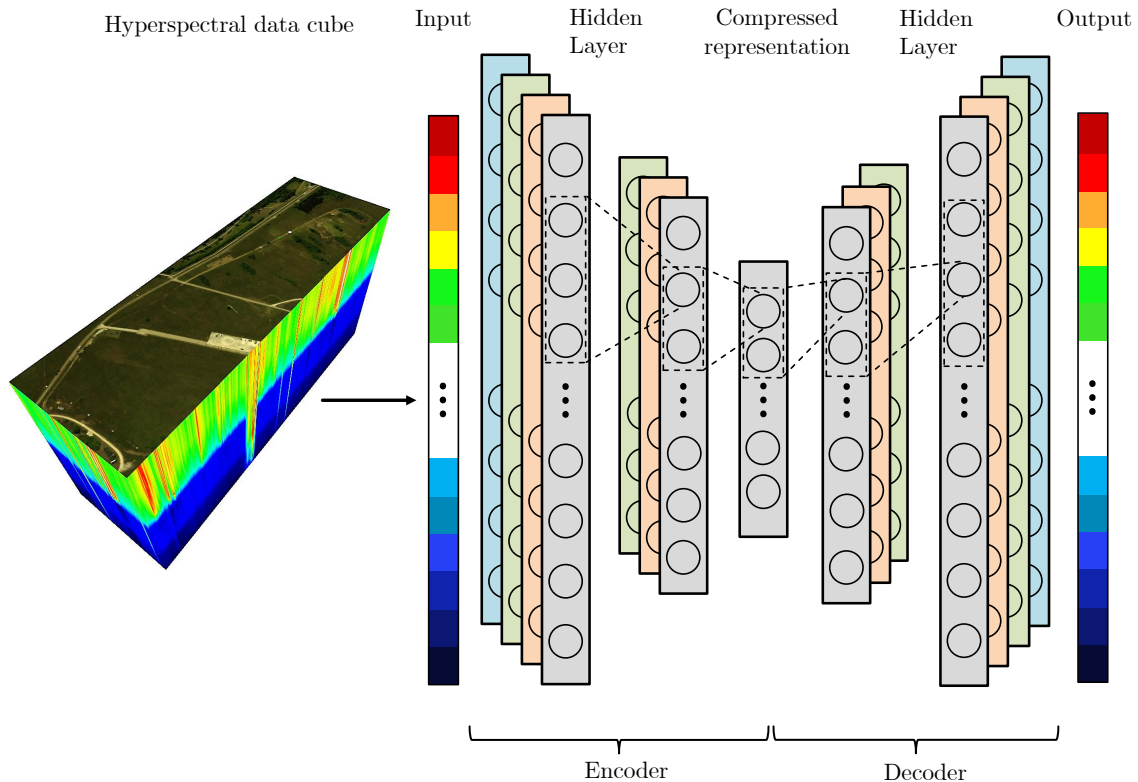
Figure 1. Individual spectral sigature from the 3D hyperspectral data cube are supplied to the 1D-CAE. In the hidden layer, several filters, shown in different colors, are used in parallel to find features and extract them. The goal is to transform the input spectral siganture into a lower dimension called compressed representation and then reconstruct the spectrum as accurately as possible.

in figure 1. The layer from which the compressed data can be extracted, is called the bottleneck (BN). The feature space is determined by the number of features in the BN and the number of filters. The compression ratio in the BN is then calculated as

$$c_{\mathrm{R}} = \frac{b \cdot F_{\mathrm{in}}}{\hat{b} \cdot \hat{F}_{\mathrm{BN}}}, \tag{1}$$

where b and $\hat{b}$ denote the number of features and $F_{\mathrm{in}}, \hat{F}_{\mathrm{BN}}$ the number of filter in the input and in the BN layer.

The decoder has the reverse encoder structure. For this the max pooling layers are replaced by upsampling layers. Two 1D-convolutional layers are added to the decoder structure. The 1D-convolutions are used for finding local features from the compressed data by increasing the number of filters that are used in parallel. After the two 1D-convolutional layers follows an upsampling layer. The upsampling layers increase the number of features by repeating the layer's input element-wise along the time axis. Then a set consisting of one 1D-convolutional layer and an upsampling layer is added to the decoder structure. The layers increase the number of features as well as the number of filters. The last 1D-convolutional layer reduces the number of filters to 1. The decoder is finalized with a cropping layer, which removes the input signal's first element to resemble the original input signal's length, if the original signal dimensionality was odd.

The 1D-convolutional layer of the network utilizes ϕ_1 as activation function, which is the Leaky Rectified Linear Unit (Leaky ReLU), to prevent the vanishing gradient problem (Goodfellow et al., 2016). For the last 1D-convolutional layer, the sigmoid function is used as the activation function ϕ_2. The sigmoid function normalizes the output of the 1D-CAE to the range of values $[0, 1] := \{\tilde{a} \in \mathbb{R} \mid 0 \leq \tilde{a} \leq 1\}$. The normalization is carried out because the input signal has a value range from 0 to 1. This is necessary to compare the reconstructed signal with the input signal and calculate the error between the two spectral signatures. Adding non-linear properties due to the activation functions ϕ_1 and ϕ_2 to the network allows the 1D-CAE model to approximate non-linear functions. This is necessary to obtain an accurate reconstruction of the non-linear feature variations in hyperspectral data (Gross et al., 2019).

For the training process, we initialize the weight matrices with random values. During the training, the weight matrices are iteratively updated to minimize the error between the original and reconstructed signal. The loss function $d(\mathbf{x}, \tilde{\mathbf{x}})$ measures the difference between the input $\mathbf{x}$ and the reconstructed signal $\tilde{\mathbf{x}}$. To evaluate reconstruction accuracy of the training process, we use the mean squared error (MSE) as loss function $d(\mathbf{x}, \tilde{\mathbf{x}})$, which can be written as

$$\epsilon_{\mathrm{MSE}}(\mathbf{x}, \tilde{\mathbf{x}}) = d(\mathbf{x}, \tilde{\mathbf{x}}) = \frac{1}{b} \sum_{i=1}^{b} (\mathbf{x}_{\mathrm{i}} - \tilde{\mathbf{x}}_{\mathrm{i}})^2. \tag{2}$$

After every epoch of the training process, each layer's weight matrix are updated using the backpropagation algorithm in combination with the Adam optimizer (Kingma and Ba, 2014). The convergence of the loss function towards a local minimum indicates a successful training process. Since this is not a convex optimization problem, the convergence to a global minimum cannot be guaranteed.

3. EXPERIMENTAL SETUP

To evaluate the reconstruction accuracy of the 1D-CAE, the two hyperspectral data sets *Greding Village* (Gross et al., 2019) and *Pavia University* (Plaza et al., 2006) are used. Both hyperspectral data sets have the shape of $\mathbf{H} \in \mathbb{R}^{\mathrm{m} \times \mathrm{n} \times \mathrm{b}}$ with b spectral bands and spatial dimensions $m \times n$. As neither the

	Layer type	Activation function	kernel size	features filter
Encoder	-	-	-	$b, 1$
	1D Zero padding	-	-	$\tilde{b} = \begin{cases} b, & \text{if } b \text{ even} \\ b+1, & \text{if } b \text{ odd} \end{cases}, 1$
	1D Convolutional	$\phi_1 =$ Leaky ReLU	11	$\tilde{b}, 64$
	1D MaxPooling	-	2	$\lfloor \tilde{b}/2 \rfloor, 64$
	1D Convolutional	$\phi_1 =$ Leaky ReLU	11	$\lfloor \tilde{b}/2 \rfloor, 32$
	1D MaxPooling	-	2	$\hat{b}, 32$
	1D Convolutional	$\phi_1 =$ Leaky ReLU	9	$\hat{b}, 16$
	1D Convolutional	$\phi_1 =$ Leaky ReLU	7	$\hat{b}, 1$
Decoder	1D Convolutional	$\phi_1 =$ Leaky ReLU	7	$\hat{b}, 16$
	1D Convolutional	$\phi_1 =$ Leaky ReLU	9	$\hat{b}, 32$
	1D UpSampling	-	2	$\hat{b} \cdot 2, 32$
	1D Convolutional	$\phi_1 =$ Leaky ReLU	11	$\hat{b} \cdot 2, 64$
	1D UpSampling	-	2	$\hat{b} \cdot 4, 64$
	1D Convolutional	$\phi_2 =$ sigmoid	11	$\hat{b} \cdot 4, 1$
	Cropping1D	-	-	$b = \begin{cases} \hat{b} - 1, & \text{if } b \text{ even} \\ \hat{b}, & \text{if } b \text{ odd} \end{cases}, 1$

Table 1. Structure of the 1D-Convolutional Autoencoder model

1D-CAE model nor the evaluation metrics use spatial neighborhood information, the 3D hyperspectral data set can be reorganized as a 2D matrix with the following transformation $\mathbf{H} \in \mathbb{R}^{m \times n \times b} \rightarrow \mathbf{V} \in \mathbb{R}^{p \times b}$, with $p = m \cdot n$.

The *Greding* data set was recorded in 2014 over Greding, Germany. It was recorded with an aisaEAGLE II hyperspectral sensor with $b = 127$ bands, covering the electromagnetic spectrum from $390 - 990\ nm$. The data was radiometrically and atmospherically corrected and georeferenced with a ground sampling distance of $0.5\ m$. A subset of the data with 670×606 pixel, which shows a part of the village of Greding, is removed from the data and exclusively used for the evaluation process. This subset is denoted as *Greding Village* for the remainder of the paper. The *Greding Village* scene includes rural areas with several vegetation types, as well as roads and residential buildings. The remaining data with approximately $1.9 \cdot 10^6$ pixel is divided into training and validation data for the 1D-CAE model.

The second benchmark hyperspectral data set is the *Pavia University* data (Plaza et al., 2006). It was acquired by the German Aerospace Centre (DLR) within the scope of the HySens project. The data set was recorded with the ROSIS-03 sensor and consists of 340×610 pixels with $b = 103$ bands. It depicts the University of Pavia's Engineering School in Italy. The scene includes a diverse area of vegetation, buildings and infrastructure. The Pavia data is also pre-processed to reflectance with a ground sampling distance of $1.3\ m$.

The training and evaluation process for the 1D-CAE model is carried out individually for both data sets, since the data have a different number of bands. However, all bands are used for training. The *Greding* and *Pavia University* data are randomly subdivide in 80% training data and 20% validation data. The *Greding Village* data is only used for the evaluation process, and thus, unknown to the corresponding 1D-CAE model. The training process parameters were slightly adjusted depending on the data set used due to the varying amount of available training samples. For *Greding*, the signal's input dimension is $b = 127$, and the number of features in the BN is $\hat{b} = 32$. The signal's input dimension of *Pavia University* is $b = 103$ and the low-dimensional feature space has $\hat{b} = 26$ features. The parameter α of the activation function ϕ_1 is set to $\alpha = 0.3$. The com-

pression rate c_R from equation (1) is calculated from b and $\hat{b}$, because $F_{in} = \hat{F}_{BN} = 1$. This results in a compression factor of $c_R \approx 4$. The number of epochs is set to 150 and the batch size to 256. The learning rate for both data sets starts with a value of $\eta = 0.001$ and is multiplied by the factor 0.9 after seven epochs in which the value of the loss function has not changed by more than $\Delta\epsilon_{MSE} = 1.0e - 07$. The learning rate is reduced to ensure convergence to a local minimum. If a minimum is reached, which until then has the lowest MSE value, the associated weight matrices of the model are saved. The weight matrix of the 1D-CAE for *Greding* and *Pavia University* are initialized with values generated by a Gaussian distribution with $\mu = 0$ and standard deviation $\sigma = 0.05$.

To evaluate reconstruction accuracy of the 1D-CAE model, the results are compared with the DAE and the NLPCA method on the reconstructed data. The DAE model was introduced in (Kuester et al., 2020) and shows a high level of reconstruction accuracy, assessed using various metrics, and in target detection under challenging conditions even for sub-pixel targets. The current state-of-the-art method NLPCA from (Licciardi and Chanussot, 2018) was re-implemented. The basic structure, as well as the activation function of the individual layers, are identical to the NLPCA method from (Licciardi and Chanussot, 2018). However, the optimization method was changed from the original conjugate gradients method to the Adam Optimizer. The batch size was increased from 1 to 256, and the number of epochs was reduced from the original 2500 to 150. The *Signal-to-Noise Ratio* (SNR) and the *Spectral Angle Mapper* (SAM) are used to measure the 1D-CAE model's reconstruction accuracy. The evaluation algorithms are applied to the original data and the reconstructed data of the different compression methods. The SNR measures the accuracy of the reconstructed data in decibels (dB) (Fowler and Rucker, 2007). The SNR is defined as

$$\epsilon_{SNR}(\mathbf{V}, \tilde{\mathbf{V}}) = \frac{1}{p} \sum_{i=1}^{p} 10 \log_{10} \frac{\boldsymbol{\sigma}_i(\mathbf{V})^2}{\boldsymbol{\epsilon}_{MSE,i}(\mathbf{V}, \tilde{\mathbf{V}})}, \quad (3)$$

where the MSE is calculated according to equation (2). The variable $\mathbf{V}$ indicates the original data and $\tilde{\mathbf{V}}$ the reconstructed

data. $\boldsymbol{\epsilon}_{\mathrm{MSE}} \in \mathbb{R}^{\mathrm{b}}$ is the mean square error between the original and the reconstructed data, and $\boldsymbol{\sigma}(\mathbf{V})^2 \in \mathbb{R}^{\mathrm{b}}$ is the variance of the original data over all pixels for each band. The smaller the difference between the original and the reconstructed data set, the larger the SNR value and the better the spectral similarity. The second metric used to calculate the similarity between the input and the reconstructed spectral signatures is the SAM (De Carvalho and Meneses, 2000). The SAM calculates the spectral angle between two spectra and is defined as

$$\boldsymbol{\epsilon}_{\mathrm{SAM}}(\mathbf{V}, \tilde{\mathbf{V}}) = \arccos \frac{\sum\limits_{i=1}^{b} \mathbf{V}_i \tilde{\mathbf{V}}_i}{\sqrt{\sum\limits_{j=1}^{b} \mathbf{V}_j{}^2 \sum\limits_{l=1}^{b} \tilde{\mathbf{V}}_l^2}} \in \mathbb{R}^{\mathrm{p}}. \quad (4)$$

The result of the calculation is a vector with one value per spectral signature in $\mathbf{V}$. The smaller each element's value, the higher the spectral similarity, which means less deviation between the original and the reconstructed signal. To compare the results, the mean SAM value is computed and used as an indicator of the deviation from the original data. Finally, the land cover classification is carried out using an SVM. For this the libSVM MATLAB package is used as a multi-class SVM with optimized step size selection (Chang and Lin, 2011). The SVM is trained exclusively on the original data sets and then applied to the reconstructed data for the land cover classification. The results are evaluated by comparing the and Cohen's Kappa to the land cover results of the original data. For the SVM training process, the original *Greding Village* and *Pavia University* data sets are used.

4. RESULTS AND DISCUSSION

The reconstruction accuracy of the 1D-CAE outperforms the DAE and NLPCA for every used metric for the *Greding Village* data set as well as *Pavia University* data set. The focus lies on the compression of the spectral signature for a fixed compression ratio of $\approx 4:1$. Table 2 shows the reconstruction accuracy measured by the SNR and the SAM for the *Greding Village* data set. The SVM classification results were calculated by applying the previously trained SVM model to the reconstructed data. The baseline indicates the hypothetical results for a perfect (loss-less) reconstruction. The 1D-CAE model surpasses the DAE by $2.9\,dB$ and the NLPCA method by $7.1\,dB$, shown in Table 2. Furthermore, the spectral angle of the 1D-CAE model is smaller compared to the two other methods. With the land cover classification, the compression with the 1D-CAE model leads to a higher classification accuracy compared to the other methods. The classification result of the proposed approach almost reaches the results of the baseline. Overall, this concludes that the 1D-CAE compression method achieves the highest reconstruction accuracy among the tested methods on the *Greding Village* data.

The corresponding evaluation of the *Pavia University* data in Table 2 shows that the 1D-CAE model achieves an SNR value that is $5.0 - 5.5\,dB$ higher compared to the other two methods. Furthermore, the spectral angle of the 1D-CAE approach is $\approx 40\,\%$ smaller than the spectral angle of the DAE and NLPCA method. In the land cover classification, the proposed approach achieves an overall classification that is $4\,\%$ higher than the DAE and NLPCA method. The overall accuracy of the 1D-CAE model is less than $2\,\%$ lower than that of the baseline.

Table 2. Comparison of the different compression methods based on SNR, SAM, and land cover classification. The baseline in the second row indicates the hypothetical results for a perfect (loss-less) reconstruction. A high SNR value and a small spectral angle indicate a good reconstruction accuracy of the original data. The best values for each metric are highlighted in bold.

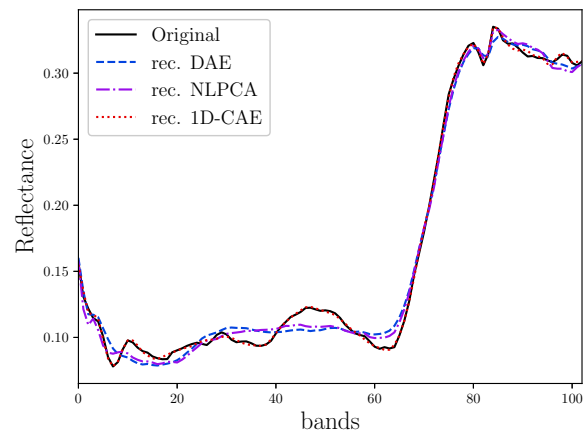| Greding Village			
Compression method	SNR in dB	SAM in degree	SVM classification
baseline	∞	0	OA: 0.964 κ: 0.955
1D-CAE	**42.79**	**0.266**	OA: **0.963** κ: **0.954**
DAE	39.89	0.311	OA: 0.960 κ: 0,950
NLPCA	35.70	0.471	OA: 0.958 κ: 0.948
Pavia University			
Compression method	SNR in dB	SAM in degree	SVM classification
baseline	∞	0	OA: 0.779 κ: 0.717
1D-CAE	**33.21**	**0.714**	OA: **0.765** κ: **0.699**
DAE	27.72	1.180	OA: 0.717 κ: 0.639
NLPCA	28.14	1.189	OA: 0.717 κ: 0.638

Figure 2. Comparison of the original spectrum of vegetation with the reconstructions of the 1D-CAE, DAE and NLPCA.

The *Pavia University* data set's reconstruction accuracy is generally lower compared to the *Greding Village* data set. This can be explained by the lower number of classes and the higher number of available training samples per class in the *Greding Village* data set, which is an important factor in training neural networks.

Figure 2 shows the original spectral signature of a vegetation sample in the *Pavia University* data set compared to the corresponding reconstructed spectra of the proposed 1D-CAE, the DAE and the NLPCA model. There are small differences in the reconstruction accuracy, which can be seen, e.g., for the bands 5 to 20, as shown in Figure 3 and for the bands 35 to 60, as shown in Figure 4. The reconstruction error between the original and the reconstruction of the spectral signature from the DAE and NLPCA is higher. This is especially evident for the bands from 5 to 20 and for 35 to 60 as shown in Figure 3 and Figure 4, respectively. Figure 5 shows the original and reconstructed spectral signature of a factory's roof. The dotted red line of the 1D-CAE reconstruction has the highest accordance with the original spectral signature. Especially in the areas of bands 1 to 20 and 70 to 90, the reconstruction error of the 1D-CAE is lower compared to the other methods. Figure 2 and Figure 5 confirms the results from Table 2.

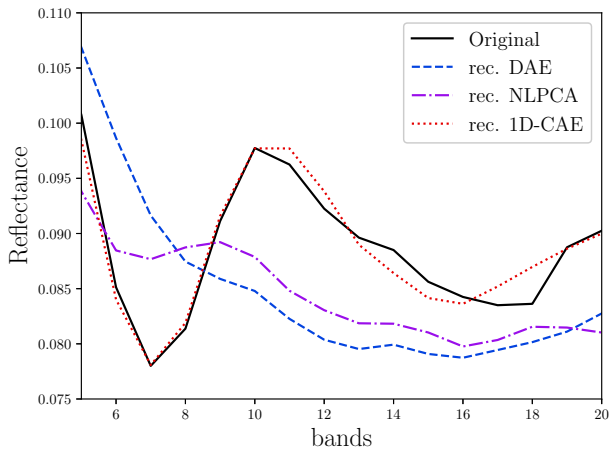The main difference between the model architecture of 1D-

Figure 3. Comparison of the original spectrum of vegetation with the reconstructions of the 1D-CAE, DAE and NLPCA in the bands 5 to 20.
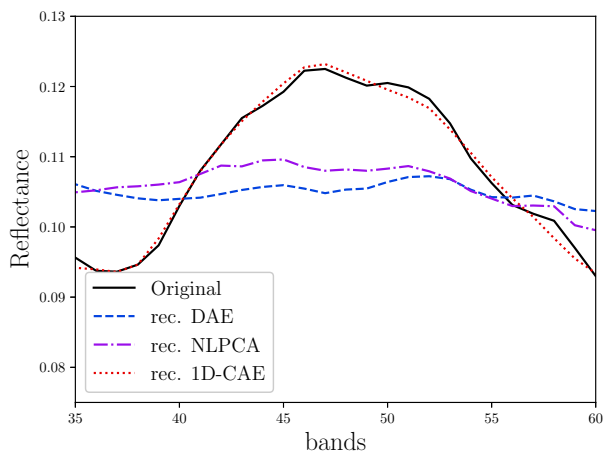

Figure 4. Comparison of the original spectrum of vegetation with the reconstructions of the 1D-CAE, DAE and NLPCA in the bands 35 to 60.
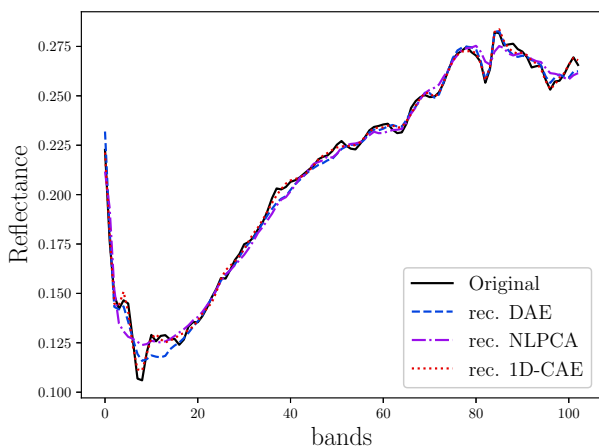

Figure 5. Comparison of the original spectrum of the factories roof with the reconstructions of the 1D-CAE, DAE and NLPCA.

CAE and the other methods are the utilized layer types. Through the use of 1D-convolutional layer with its local filter, the important features in the spectrum can be found more easily. Another advantage is that a once recognized pattern at a specific position in the spectral signature, can be automatically identified at a different position. This attribute makes the 1D-convolutions translation-invariant towards features. By using multiple filters stacked in parallel, a large number of different features can be found. The 1D-convolutional layer is followed by a max pooling layer, which is used to extract only the most significant features. Thus, the total amount of features is reduced, which results in the desired compression. Two additional benefits are a substantial reduction of the computational cost for the following layer, and the prevention of overfitting. Since the DAE and NLPCA do not use any local filters, they lack the property of translation invariance. This means that two identical features must be found separately from one another in different bands. Furthermore, they do not have filters stacked in parallel, which results in a smaller amount of features found. With the improved feature extraction of the 1D-CAE, the important features for the compression can be found. The same applies to the reconstruction from the extracted features. The focus lies on the compression of the spectral dimension. The spatial dimension is not used in the compression in order not to falsify the correlation between the spectral dimension and the accuracy of the reconstruction.

For large and deep neural network structures, a lot of training data is required to robustly find and extract the important features. In the case of convolutional networks, high reconstruction accuracies can still be achieved for small amounts of training data. This can be explained by the translation invariance of the convolutional kernels with respect to features. However, it is also advantageous to train the model with many hyperspectral signatures of different materials in order to optimize performance for all materials and their spectral variations in the data. The 1D-CAE shows a good training process because the weight matrices are optimized the most in the early epochs, which means that the MSE values are rapidly decreasing in the beginning. This behavior is observed for the training of the 1D-CAE model on both data sets. Since the *Greding Village* data were entirely unknown for the compression method, the model shows a good generalization capabilities, as the training was performed in a predominantly agricultural area while the test took place with rural scene including a small village. In the evaluation of the 1D-CAE on the *Pavia University* data set the same area is utilized for the validation and the evaluation process. This is not detrimental to the evaluation process, as the training and validation data are disjoint.

The generalization capability of the 1D-CAE is also given for the *Pavia University* data set. During the training process, the error for the validation data is in the same order of magnitude as the error for the training data, and there are no significant outliers, it is assumed that there is no oveoutliers. This indicates that the training process does not result in overfitting. At this stage, the scalability of the proposed method is limited by the fixed number of spectral bands for the Input layer. This can be solved by specific modules to adjust the spectral dimension of the input data to the model, in case of different sensor models. In conclusion, the 1D-CAE model's performance surpasses the DAE and the state-of-the-art NLPCA method and is able to efficiently and accurately compress the spectral dimension of hyperspectral data.

5. OUTLOOK

This paper proposes a 1D-CAE model to efficiently compress the spectral dimension of hyperspectral data. A high level of reconstruction accuracy was achieved and demonstrated by comparing the SNR and the spectral similarity, as well as an evaluation by land cover classification between the original and the reconstructed data for the *Greding Village* and the *Pavia University* data sets. In order to examine the overall generalization capability, data with more diverse setting have to be tested. Additionally, the impact of the correlation, which can change significantly from scene to scene, will be investigated. More test data is required for reliable results. In future research, higher compression rates will be investigated to compress the spectral dimension. It will be investigated whether the compression rate can be increased after applying additional spatial compression, while simultaneously maintaining a comparable reconstruction accuracy. The spatial dimension's compression can be carried out using a classic method such as a 2D-DWT or a 2D-AE that takes the spatial properties into account. With a 3D-convolutional neural network, spatial and spectral compression can be carried out simultaneously. This could allow a combination of spatial and spectral features, and thus improve feature extraction. Additionally, we plan to compare our results to compression methods from the field of signal processing, such as the cosine transformation and wavelet transformation.

Depending on the use-case, it can be beneficial to introduce additional evaluation metrics as loss functions to extract specific features for a desired target application. Additionally, the transfer of data sets to a compression model with a different number of bands is an important topic, because the number of input features cannot be changed. Finally, the development of a model that can handle input data of arbitrary dimensionality is an important research topic, as the current models are tailored to a specific number of input features. This would allow the compression of data from different sensors and greatly facilitate usability.

REFERENCES

Abousleman, G. P., Marcellin, M. W., Hunt, B. R., 1995. Compression of hyperspectral imagery using the 3-D DCT and hybrid DPCM/DCT. *IEEE transactions on geoscience and remote sensing*, 33(1), 26–34.

Chang, C.-C., Lin, C.-J., 2011. LIBSVM: A Library for Support Vector Machines. *ACM Trans. Intell. Syst. Technol.*, 2(3). https://doi.org/10.1145/1961189.1961199.

De Carvalho, O. A., Meneses, P. R., 2000. Spectral correlation mapper (scm): an improvement on the spectral angle mapper (sam). *Summaries of the 9th JPL Airborne Earth Science Workshop, JPL Publication 00-18*, 9, JPL Publication Pasadena, CA.

Du, Q., Fowler, J. E., 2007. Hyperspectral Image Compression Using JPEG2000 and Principal Component Analysis. *IEEE Geoscience and Remote Sensing Letters*, 4(2), 201-205.

Du, Q., Fowler, J. E., 2008. Low-complexity principal component analysis for hyperspectral image compression. *The International Journal of High Performance Computing Applications*, 22(4), 438–448.

Fowler, J. E., Rucker, J. T., 2007. Three-dimensional wavelet-based compression of hyperspectral imagery. *Hyperspectral Data Exploitation: Theory and Applications*, 379–407.

Goodfellow, I., Bengio, Y., Courville, A., 2016. *Deep Learning*. The MIT Press.

Gross, W., Tuia, D., Soergel, U., Middelmann, W., 2019. Nonlinear feature normalization for hyperspectral domain adaptation and mitigation of nonlinear effects. *IEEE Transactions on Geoscience and Remote Sensing*, 57(8), 5975–5990.

Kingma, D. P., Ba, J., 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kuester, J., Gross, W., Middelmann, W., 2020. An approach to near-lossless hyperspectral data compression using deep autoencoder. L. Bruzzone, F. Bovolo (eds), *Image and Signal Processing for Remote Sensing XXVI*, 11533, SPIE, 210 – 220.

Li, J., Liu, Z., 2019. Multispectral transforms using convolution neural networks for remote sensing multispectral image compression. *Remote Sensing*, 11(7), 759.

Licciardi, G. A., Chanussot, J., Piscini, A., 2014. Spectral compression of hyperspectral images by means of nonlinear principal component analysis decorrelation. *2014 IEEE International Conference on Image Processing (ICIP)*, 5092–5096.

Licciardi, G., Chanussot, J., 2018. Spectral transformation based on nonlinear principal component analysis for dimensionality reduction of hyperspectral images. *European Journal of Remote Sensing*, 51(1), 375-390. https://doi.org/10.1080/22797254.2018.1441670.

Lim, S., Sohn, K., Lee, C., 2001. Compression for hyperspectral images using three dimensional wavelet transform. *IGARSS 2001. Scanning the Present and Resolving the Future. Proceedings. IEEE 2001 International Geoscience and Remote Sensing Symposium (Cat. No. 01CH37217)*, 1, IEEE, 109–111.

Penna, B., Tillo, T., Magli, E., Olmo, G., 2006. A new low complexity klt for lossy hyperspectral data compression. *2006 IEEE International Symposium on Geoscience and Remote Sensing*, Citeseer, 3525–3528.

Plaza, A., Benediktsson, J. A., Boardman, J., Brazile, J., Bruzzone, L., Camps-Valls, G., Chanussot, J., Fauvel, M., Gamba, P., Gualtieri, A. et al., 2006. Advanced processing of hyperspectral images. *2006 IEEE International Symposium on Geoscience and Remote Sensing*, IEEE, 1974–1978.

Priya, S., Ghosh, R., Bhattacharya, B., 2019. Non-Linear Autoencoder Based Algorithm for Dimensionality Reduction of Airborne Hyperspectral Data. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42(3/W6).

Theis, L., Shi, W., Cunningham, A., Huszár, F., 2017. Lossy image compression with compressive autoencoders. *arXiv preprint arXiv:1703.00395*.

Zabalza, J., Ren, J., Zheng, J., Zhao, H., Qing, C., Yang, Z., Du, P., Marshall, S., 2016. Novel segmented stacked autoencoder for effective dimensionality reduction and feature extraction in hyperspectral imaging. *Neurocomputing*, 185, 1–10.