Investigating the influence of hyperspectral data compression on spectral unmixing

Jannick Kuester, Johannes Anastasiadis, Wolfgang Middelmann, Michael Heizmann

Jannick Kuester, Johannes Anastasiadis, Wolfgang Middelmann, Michael Heizmann, "Investigating the influence of hyperspectral data compression on spectral unmixing," Proc. SPIE 12267, Image and Signal Processing for Remote Sensing XXVIII, 122670H (26 October 2022); doi: 10.1117/12.2636129

Event: SPIE Remote Sensing, 2022, Berlin, Germany

Investigating the influence of hyperspectral data compression on spectral unmixing

Jannick Kuester[a,b], Johannes Anastasiadis[b], Wolfgang Middelmann[a], and Michael Heizmann[b]

[a]Fraunhofer IOSB, Ettlingen, Germany, Fraunhofer Center for Machine Learning
[b]Institute of Industrial Information Technology (IIIT), Karlsruhe Institute of Technology (KIT), Hertzstraße 16, 76187 Karlsruhe, Germany

ABSTRACT

This work addresses the problem of hyperspectral data compression and the evaluation of the reconstruction quality for different compression rates. Data compression is intended to transmit the enormous amount of data created by hyperspectral sensors efficiently. The information loss due to the compression process is evaluated by the complex task of spectral unmixing. We propose an improved 1D-Convolutional Autoencoder architecture with different compression rates for lossy hyperspectral data compression. Furthermore, we evaluate the reconstruction by applying metrics such as SNR and SA and compare them to the spectral unmixing results.

Keywords: hyperspectral data compression, spectral unmixing, feature extraction, dimensionality reduction, Autoencoder, hyperspectral imaging, spectral analysis, remote sensing, convolutional neural networks

1. INTRODUCTION

The demand for hyperspectral data in the fields of remote sensing and industrial visual inspection is continuously increasing. Applications such as disaster management,[1] land use classification,[2] object detection,[3] and spectral unmixing[4] benefit from the additional spectral information provided by hyperspectral data, which are material-specific features. Hyperspectral sensors measure the electromagnetic spectrum in up to hundreds of narrow and contiguous spectral bands. Depending on the field of application, the hyperspectral sensor is mounted on different carrier platforms. In remote sensing, typical carrier platforms are drones, aircraft, or satellites. According to the carrier system, different area coverages with different spatial resolutions can be achieved. The common problem of the different systems is, that the many spectral bands lead to an enormous amount of data, while the transmission bandwidth or storage space is limited. Both bandwidth and storage space can only be expanded at great expense, especially in satellite-based systems. Thus, it is important to solve the problem of data transmission and storage space.

Lossy data compression is one of the most promising approaches to achieve efficient data transmission. Lossy compression methods can be broadly divided into the following two categories: signal processing methods and machine learning methods. Classical signal processing methods use quantization techniques to compress the hyperspectral data.[5,6] Other algorithms rely on decorrelation transformations to exploit spectral correlation, followed by a quantization stage and an entropy encoder. For the reduction of spectral correlation, methods such as the discrete cosine transform, discrete wavelet transform (DWT), and principal component analysis (PCA) are used.[7,8] The 1D-spectral decorrelators are combined with JPEG 2000 as a spatial decorrelator.[9–11] Current literature shows that PCA-based methods outperform the DWT in terms of reconstruction accuracy and compression performance.[12,13]

Machine learning-based compression techniques show higher reconstruction accuracy compared to classical signal processing techniques in compressing RGB image data.[14,15] It has been shown that Autoencoder structures are particularly suitable for the compression of RGB images and even outperform state-of-the-art methods such as JPEG and JPEG 2000.[15,16] Due to the convincing performance of Autoencoder-based (AE) approaches in image compression, they are increasingly used for hyperspectral data processing. Initially, Autoencoder-based approaches were used for dimensional reduction of hyperspectral data and later extended to reconstruct features.[17–19] Several publications show, that Autoencoder-based methods achieve a higher reconstruction accuracy than PCA methods.[20–22] Further development stages retain the Autoencoder structure and replace the fully

Image and Signal Processing for Remote Sensing XXVIII, edited by Lorenzo Bruzzone,
Francesca Bovolo, Nazzareno Pierdicca, Proc. of SPIE Vol. 12267, 122670H
© 2022 SPIE · 0277-786X · doi: 10.1117/12.2636129

connected layers with convolutional layers to increase reconstruction accuracy.[23] Furthermore, 1D-Convolutional Autoencoder (1D-CAE) achieve high reconstruction accuracy for various datasets and error metrics.[24]

Spectral unmixing is required if several materials are contained in one hyperspectral pixel, which means that only a mixed spectrum is available. This happens when the observed material particles are small or the distance between camera and object is large, which is often the case in remote sensing measurements. With spectral unmixing, the material composition of such pixels can be examined.[25]

This work addresses the problem of hyperspectral data compression and the evaluation of the reconstruction quality for different compression rates based on the spectral unmixing application. We propose an improved 1D-CAE architecture of the base version from Kuester et al.[24] with the compression rates of $C_{\mathrm{R}} = 4$, $C_{\mathrm{R}} = 8$, and $C_{\mathrm{R}} = 16$ for hyperspectral data compression. Since the applied compression methods are lossy methods, it is important that the loss of information is assessed using metrics. It is important for a user to know the impact of the compression loss for the application to be analyzed. For this purpose, we evaluate the performance of our proposed lossy compression method on the complex task of spectral unmixing. The evaluation is performed on data from material mixtures collected in the IIIT image processing laboratory named quartz-3 and quartz-4 as well as the HYDICE Urban benchmark dataset from the remote sensing domain. All datasets consist of sample sets with known mixture coefficients. In this contribution, we will show that tere is no correlation between the signal-to-noise ratio, spectral angle, and the spectral unmixing results. Furthermore, the different compression ratios have no significant effect on the unmixing.

2. PROPOSED COMPRESSION METHOD

Our proposed method is an extension of the 1D-CAE to compress the spectral dimension of hyperspectral data. Since the majority of hyperspectral applications such as spectral unmixing use spectral information, accurate reconstruction of spectral signatures is essential. For this reason, the possibility of compressing the spatial dimension is not considered for now and the focus is on the spectral dimension. This approach ensures that the reconstruction accuracy is not negatively affected by spatial compression effects. Thus, the quality of the reconstruction accuracy can be directly related to the spectral compression. Since the information of the spatial neighborhood is left out, the 3D hyperspectral dataset can be represented by a set of N discrete reflectance spectra $\mathbf{y}_n = [y_{1n}, ..., y_{Bn}]^{\mathrm{T}} \in \mathbb{R}^B$.

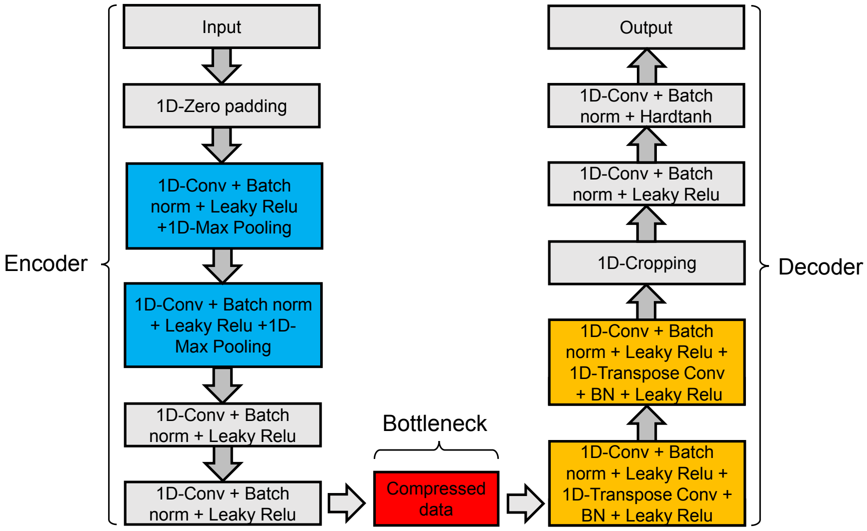
Encoder
Input
1D-Zero padding
1D-Conv + Batch norm + Leaky Relu +1D-Max Pooling
1D-Conv + Batch norm + Leaky Relu +1D-Max Pooling
1D-Conv + Batch norm + Leaky Relu
1D-Conv + Batch norm + Leaky Relu
Bottleneck
Compressed data
Decoder
Output
1D-Conv + Batch norm + Hardtanh
1D-Conv + Batch norm + Leaky Relu
1D-Cropping
1D-Conv + Batch norm + Leaky Relu + 1D-Transpose Conv + BN + Leaky Relu
1D-Conv + Batch norm + Leaky Relu + 1D-Transpose Conv + BN + Leaky Relu

Figure 1. The basic structure of the 1D-CAE with a compression rate of four. Replicating the blue block increases the compression rate by a factor of two. The orange blocks, inserted the same number of times as the blue ones, restore the number of original features.

The Autoencoder structure has a fixed compression rate based on its components and has to be adapted to produce different compression rates. Figure 1 shows the structure of the 1D-CAE for the compression rate

$C_{\mathrm{R}} = 4$. The 1D-CAE model can be divided into an encoder and a decoder part. This also offers the advantage of operating both components independently on different platforms. The encoder takes a single spectral signature $\mathbf{y}$ as input and extracts the essential features during the compression process.

The Autoencoder structure proposed here also depends on the number of bands of the spectral signature. Each band will be considered as a separate feature for the remainder of this paper. The input layer is followed by a zero-padding layer that inserts features at the beginning or at the end of the spectrum such that the total number of features can be divided by the previously selected compression rate without a residuum. The 1D-convolutional layer has the purpose of recognizing and extracting local features from the spectral signature. To find a variety of different features, we use multiple filters in parallel. The 1D-convolutional layer is followed by a batch normalization layer, which is used especially for deeper networks to counteract the attenuation of the internal covariate shift and to speed up the training as well as to generate more reliable models.[26] Subsequently, the Leaky Rectified Linear Unit (Leaky ReLU) activation function is applied to the output of the batch normalization layer. We choose the Leaky ReLU as activation function for the hidden layers to avoid vanishing gradients for negative inputs of the activation functions. The Leaky ReLU has a low slope for negative values, which is set with the parameter α. Using non-linear activation functions allows an approximation of non-linear functions. Afterwards, the max pooling layer calculates the maximum value of a local area of the feature map from the output of the Leaky ReLU layer. Max pooling is used as a down-sampling operation to reduce the dimensionality while simultaneously preserving the characteristic features. With each max pooling layer, the number of features is reduced by two. These four layers can be combined to form a module, as shown by the blue box in Figure 1. The compression rate is determined by how often this module occurs consecutively in the Autoencoder structure. The subsequent 1D-convolutional layers reduce the parallel filters towards the bottleneck (BN). The individual transformations of each layer are concatenated into an overall transformation for the encoder $\mathbf{f}$. The encoder transforms the spectral signature to a low-dimensional representation

$$\mathbf{h} = \mathbf{f}(\mathbf{W}_{\mathrm{E}}, \mathbf{y}) \in \mathbb{R}^{\hat{B}}, \tag{1}$$

where $\mathbf{W}_{\mathrm{E}}$ indicates the encoder weights. The lossy compression is realized by transforming the input spectral signature into a low-dimensional representation, using only $\hat{B} \ll B$ features in the BN to represent the spectral information. The compression rate can be calculated as

$$C_{\mathrm{R}} = \frac{B \cdot F_{\mathrm{in}}}{\hat{B} \cdot \hat{F}_{\mathrm{BN}}}, \tag{2}$$

where B and F_{in} denote the features and feature maps of the input layer and $\hat{B}$ and $\hat{F}_{\mathrm{BN}}$ the number of features and feature maps in the BN. The number of feature maps is one in the input layer as well as in the BN. The compression rate depends entirely on the number of features in the BN, which is controlled by the number of pooling layers. The compressed signal can be extracted from the BN and transmitted to the decoder with the corresponding transformation $\mathbf{g}$. The decoder transforms the compressed spectral information back to the original representation according to

$$\tilde{\mathbf{y}} = \mathbf{g}(\mathbf{W}_{\mathrm{D}}, \mathbf{h}) \in \mathbb{R}^{B}, \tag{3}$$

where $\mathbf{W}_{\mathrm{D}}$ are the decoder weights. The task of the decoder is to reconstruct the input spectral signature as accurately as possible. To increase the accuracy of the transformation of the decoder, the 1D-Transposed Convolutional Layer (1D-TCL) is used as the upsampling operation instead of the upsampling layer. Compared to the static upsampling method used so far, the 1D-TCL is trainable and more adaptable. The parameters of the 1D-TCL are set to increase the number of features by a factor of two. Each 1D-TCL is preceded by a 1D-convolutional layer whose task is to generate new features between the already existing ones. Together with the batch normalization and the activation function, equivalent to the encoder, they can be combined to modules, depicted as orange boxes in Figure 1, which increases the number of features. The cropping layer removes features at the beginning and at the end to restore the original number of features. The subsequent 1D-convolutional layers are used to reduce the number of feature maps towards the output of the decoder. The final layer of the 1D-CAE architecture utilizes a Hardtanh activation, limiting the output value range to [0,1]. Since the mean-squared-error $epsilon_{\mathrm{MSE}}$ is used as the loss function to calculate the reconstruction error, it is necessary that the input and output value range is equal.

Another advantage of the Autoencoder structure is that the input spectral signature is also used as a reference to evaluate the reconstruction accuracy. This results in a semi-supervised learning paradigm without the requirement for additional labelled data. Now it is to be examined how the presented compression method affects spectral unmixing. For this purpose, the performance of spectral unmixing with different methods will be compared between the uncompressed data and the compressed data. In the next section spectral unmixing methods will be introduced.

3. SPECTRAL UNMIXING

This section introduces spectral unmixing and presents established methods that will later be used to evaluate the compression results. Readers who are familiar with spectral unmixing can skip this section.

The estimation of relative material proportions (abundances) in a mixture of substances based on its spectrum is called spectral unmixing.[25] If the spectra of the pure substances are known, as it is the case with the datasets used in this paper, it is referred to as supervised spectral unmixing. The estimation $\hat{\mathbf{a}} = [\hat{a}_1, ..., \hat{a}_P]^{\mathrm{T}} \in \mathbb{R}^P$ of the abundances $\mathbf{a} = [a_1, ..., a_P]^{\mathrm{T}} \in \mathbb{R}^P$ of the up to P pure substances involved must fulfill constraints in order to preserve physical plausibility: The non-negativity constraint

$$\hat{a}_p \geq 0 \quad \forall p \tag{4}$$

and the sum-to-one constraint

$$\sum_{p=1}^{P} \hat{a}_p = 1\,. \tag{5}$$

To perform spectral unmixing, an inverse problem has to be solved, which is often achieved using a mixing model.[27] A straightforward mixing model that yields valid results for macroscopic scale mixtures is the linear mixing model (LMM):[28,29]

$$\mathbf{y} = \sum_{p=1}^{P} \mathbf{m}_p \, a_p = \mathbf{M}\,\mathbf{a}\,. \tag{6}$$

Here, $\mathbf{y} = [y_1, ..., y_B]^{\mathrm{T}} \in \mathbb{R}^B$ is a discrete reflectance spectrum of a mixture of substances sampled at B spectral bands. The matrix $\mathbf{M} = [\mathbf{m}_1, ..., \mathbf{m}_P] \in \mathbb{R}^{B \times P}$ contains the pure spectra of the up to P substances involved.

When light propagates along multiple paths, interacting with up to two pure substances on one path, the mixing behavior can be modeled with a bilinear model. This is the case with the HYDICE Urban benchmark dataset. In this work, the linear-quadratic mixing model (LQM)

$$\mathbf{y} = \sum_{p=1}^{P} \mathbf{m}_p \, a_p + \sum_{p=1}^{P} \sum_{q=1}^{P} \xi_{pq} \, \mathbf{m}_p \odot \mathbf{m}_q \tag{7}$$

published for spectral unmixing in urban areas is used.[30] Here $\xi_{pq} \in \mathbb{R}$ are the nonlinearity coefficients determined together with the abundances a_p.

When mixing takes place at a microscopic scale, as in the case of the colored quartz sands considered in this work (see Section 4.1), the mixing is best described in the albedo domain. The single scattering albedo $\boldsymbol{\omega} \in \mathbb{R}^B$ describes the power scattered by a particle in relation to the incident power. The transformation $R : \mathbb{R} \to \mathbb{R}$ between albedo and reflectance depends on the material properties and the imaging conditions.[31] In this paper, the transformation for the relative reflectance

$$y_b = R(\omega_b) = \frac{\omega_b}{(1 + \iota_0\sqrt{1 - \omega_b})(1 + \iota\sqrt{1 - \omega_b})} \tag{8}$$

is used, where ι_0 and ι are the cosines of the incident and scattering angles, respectively. In equation (8), isotropic scattering particles, a porosity coefficient of 1 and a negligible opposition effect are assumed. Despite these simplifying assumptions, satisfying results can be achieved in practice.[32–34] Assuming equal mean particle sizes and densities, the mixing in the albedo domain can be described by the LMM, which can be used for spectral unmixing.[32–34]

A commonly used unmixing method based on the LMM considering the constraints (4) and (5) is the *Fully-Constrained-Least-Squares* method (FCLS).[35] Here, the Lagrangian function $L : \mathbb{R}^{P+1} \to \mathbb{R}$ with

$$L(\mathbf{a}, l) = \|\mathbf{y} - \mathbf{M}\,\mathbf{a}\|_2^2 - l\left(\sum_{p=1}^{P} a_p - 1\right) \tag{9}$$

is minimized by tuning the abundances and the Lagrange multiplier $l \in \mathbb{R}$. It is an iterative procedure in which negative $\hat{a}_p$ and the corresponding pure spectra in $\mathbf{M}$ are removed. If the spectral unmixing is carried out in the albedo domain, only $\mathbf{y}$ must be replaced by $R^{-1}(\mathbf{y})$ and $\mathbf{M}$ by $R^{-1}(\mathbf{M})$. Note that the function R^{-1} is applied to the individual elements of the vector or matrix.

So far, it is assumed that spectra of identical abundances are identical. However, this is not the case in reality. Instead, spectral variability occurs.[36,37] Spectral variability has many causes, such as surface topology and the angles between the light source, object, and the camera (see (8)) or atmospheric effects.[38] There are methods that take spectral variability into account, such as the Extended Linear Mixing Model (ELMM):[39]

$$\mathbf{y} = \mathbf{M}\,\mathbf{\Phi}\,\mathbf{a}\,. \tag{10}$$

Here, $\mathbf{\Phi} \in \mathbb{R}^{P\times P}$ is a diagonal matrix. The diagonal elements operate as scaling factors of the pure spectra. With this, differences in illumination and topography can be modeled. The additional parameters are determined together with $\hat{\mathbf{a}}$ in an optimization process. Spectral unmixing is used as an application to evaluate the compression performance of the methods presented in the next section.

4. EXPERIMENTAL SETUP

4.1 Datasets

To evaluate the different compression methods by spectral unmixing, we use different real datasets. Two of them were acquired in the image processing lab of IIIT.[40] Thus, we precisely know the abundances. The datasets consist of mixtures of colored quartz sands. The spectra of the datasets consist of 91 spectral bands from 450 nm to 810 nm with an average width of 4 nm. Of each mixture, 400 spectra were captured. The first dataset contains 45 mixtures of at most 3 different colored quartz sands (quartz-3), which vary in abundance steps of 0.125. The second one contains 56 mixtures of at most 4 different colored quartz sands (quartz-4), which vary in abundance steps of 0.2. Both datasets include every abundance vector $\mathbf{a}$ that is possible with the given step size and does not violate the constraints (4) and (5).

From the remote sensing domain, we use the widely used HYDICE Urban hyperspectral dataset.[41,42] The dataset consists of 307 × 307 pixels and, thus, 94249 spectral signatures are available as samples. Each spectral signature has 210 bands and covers the wavelength range from 400 nm to 2500 nm with a spectral resolution of 10 nm. Of the 210 bands, a total of 48 bands were removed due to dense water vapor and atmospheric effects, leaving 162 bands available for experiments. The 4 pure substances of the spectral unmixing ground truth are respectively asphalt, grass, tree, and roof.

4.2 Evaluation Metrics

As an evaluation metric for spectral unmixing, we use the root-mean-square error $\epsilon_{\mathrm{RMSE}} \in \mathbb{R}$ of the abundances over a whole test dataset, that contains N samples:

$$\epsilon_{\mathrm{RMSE}} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\frac{1}{P}\sum_{p=1}^{P}(\hat{a}_{pn} - a_{pn})^2}\,. \tag{11}$$

To determine the information loss caused by compression, we evaluate the reconstruction accuracy using the signal-to-noise ratio[43] (SNR) and Spectral Angle[44] (SA) metrics. We aim to find correlations between the two error measures SNR and SA and the results from spectral unmixing to better evaluate the reconstruction accuracy. The evaluation algorithms are applied to the original and the reconstructed spectral signatures of the different compression methods and compressions rates. The SNR is defined as

$$\epsilon_{\mathrm{SNR}} = \frac{1}{B} \sum_{b=1}^{B} 10 \log_{10} \frac{\sigma_b^2}{\epsilon_{\mathrm{MSE},b}} \tag{12}$$

with the estimated variance per band

$$\sigma_b^2 = \frac{1}{N-1} \sum_{n=1}^{N} \left(y_{bn} - \bar{y}_b\right)^2 \tag{13}$$

and the mean squared error per band

$$\epsilon_{\mathrm{MSE},b} = \frac{1}{N} \sum_{n=1}^{N} \left(y_{bn} - \tilde{y}_{bn}\right)^2 . \tag{14}$$

Here, $\bar{y}_b$ is the mean band value over all spectra in the dataset. The second metric used to calculate the similarity between the input and the reconstructed spectral signatures is the SA. We use the mean SA over all N samples of a dataset:

$$\epsilon_{\mathrm{SA}} = \frac{1}{N} \sum_{n=1}^{N} \arccos \frac{\mathbf{y}_n^{\mathrm{T}} \tilde{\mathbf{y}}_n}{\|\mathbf{y}_n\| \cdot \|\tilde{\mathbf{y}}_n\|} . \tag{15}$$

According to Licciardi et al.[20] an SA below 3° indicates a high level of reconstruction accuracy.

4.3 Comparison Compression Method

To evaluate the reconstruction accuracy of the 1D-CAE, comparative lossy compression techniques that are also based on machine learning are used in addition to the evaluation metrics. The first method is the state-of-the-art NLPCA,[20] which was re-implemented. The basic structure, as well as the activation function of the individual layers have remained the same. The optimization method was changed from the original conjugate gradient method to the Adam Optimizer,[45] which increases the performance. The second comparison method is the Deep Autoencoder (DAE).[46] It is a deep neural network underlying an Autoencoder structure. It uses only fully connected layers to reduce the spectral dimension.

4.4 Hyperparameter selection

For each of the three compression rates $C_{\mathrm{R}} = 4$, $C_{\mathrm{R}} = 8$, and $C_{\mathrm{R}} = 16$, a separate model is created and trained. The compression rate depends on the number of features and feature maps from the input and the BN spectral signature, as shown in Equation (2). The number of features corresponds to the bands of the input layer, which are given by the sensor. As described in Section 2, a spectral signature is passed to the 1D-CAE as input, which leads to $F_{\mathrm{in}} = 1$. Furthermore, the number of feature maps in the BN is set to $\hat{F}_{\mathrm{BN}} = 1$ to obtain a compressed representation of the input signal. As a result, the compression rate depends entirely on the number of input and BN features. Table 1 shows the number of features required in the BN to achieve the desired compression rate of $C_{\mathrm{R}} = 4$, $C_{\mathrm{R}} = 8$, and $C_{\mathrm{R}} = 16$.

Table 1. Overview of the features required in the BN to achieve the desired compression rate for the different datasets.

Compression rate C_{R}	4	8	16
BN features for HYDICE Urban	41	21	11
BN features for quartz-3	23	12	6
BN features for quartz-4	23	12	6

The training and evaluation process for the compression models is carried out individually for the HYDICE Urban and quartz-4 dataset since the datasets have a different number of bands. Both datasets are randomly subdivided in 70 % training data, 15 % validation data, and 15 % test data. Since quartz-4 has the same number of bands and contains similar spectra as quartz-3, the complete dataset is used as test data for the network trained with the quartz-4 training dataset. This also allows the verification of the transferability of the compression methods. The following parameter set for the training process is equal for both datasets and each compression model.

The number of epochs is set to 200 and the batch size to 32. The learning rate starts with 0.001 and is multiplied by the factor 0.9 after five epochs in which the value of the loss function has not changed by more than $\Delta\epsilon_{\mathrm{MSE}} = 10^{-7}$. The learning rate is reduced to ensure convergence to a local minimum. The weight matrices of the compression methods are initialized with values generated by a Gaussian distribution. The weights of the model in the epoch that gives the smallest error ϵ_{MSE} are stored. For the Leaky ReLU activation function in the 1D-CAE model, $\alpha = 0.3$ is selected.

4.5 Spectral unmixing

For evaluation of spectral unmixing, the results of the two best performing methods from Section 3 for each dataset are presented. These are for the datasets quartz-3 and quartz-4 the FCLS algorithm and spectral unmixing based on ELMM. In both cases, spectral unmixing is performed in the albedo domain (FCLS-AD and ELMM-AD). It's no surprise that spectral unmixing in the albedo domain works best, as these are fine powders where mixing happens on a microscopic scale. As these datasets were captured in a laboratory, the incident and scattering angles are approximately known here and set to 30° and 0°, respectively.

For the HYDICE Urban hyperspectral dataset, ELMM-AD also works best, with the LQM being second. Since the angles for the transformation into the albedo domain are not known, they were assumed and set to the same values as for the other datasets. It is not necessary to know the exact angles here, because the transfomation changes significantly only for larger angles due to the cosine.

To determine abundances, the FCLS from Equation (9) was used as described in Section 3. For spectral unmixing based on the ELMM, $\mathbf{a}$ and $\mathbf{\Phi}$ are alternately minimized, where $\mathbf{a}$ is initialized with the result of the FCLS. If spectral unmixing is performed in the albedo domain, all spectra are transformed into the albedo domain beforehand and unmixed there, using FCLS and ELMM based unmixing, respectively. For spectral unmixing based on the LQM, gradient-based optimization is used to determine $\mathbf{a}$ and all ξ_{pq} simultaneously. Again, $\mathbf{a}$ is initialized with the result of the FCLS algorithm.

5. EXPERIMENTAL RESULTS

We evaluate the reconstruction accuracy of the different compression methods and compression rates using the SA, SNR, as well as the spectral unmixing. The datasets introduced above from industrial applications and from remote sensing are used. The results of the evaluation based on the SA and SNR metrics for datasets HYDICE Urban, quartz-3, and quartz-4 are shown in Table 2. The best results are printed in bold. The proposed 1D-CAE method outperforms the DAE and NLPCA for $C_{\mathrm{R}} = 4$ and $C_{\mathrm{R}} = 8$ and for all datasets. Furthermore, the reconstruction accuracy of 1D-CAE is highest for compression rate $C_{\mathrm{R}} = 16$ on datasets HYDICE Urban and quartz-4. For the HYDICE Urban dataset, the SA of the 1D-CAE is below 1° and SNR is above 100 dB for all compression rates, which indicates a very high reconstruction accuracy.

For the quartz-4 dataset, the SNR values are slightly lower and SA values higher than for the HYDICE Urban dataset, but show an acceptable accuracy. An interesting effect occurs with the quartz-3 data. Here, it can be seen that the SA values are lower compared to quartz-4 and the SNR values are higher. This correlation can be explained by the fact that the spectral signatures are smoothed during reconstruction. These smoothing effects can have positive effects on the SNR metric. In the case of unknown datasets, such as quartz-3, smoothing effects occur more frequently, which can explain the differences to quartz-4. Only for $C_{\mathrm{R}} = 16$ on quartz-3 the accuracy of the 1D-CAE is lower compared to the reference methods. One possible explanation for this is that the transferability of the filter weights found for the quartz-3 dataset is low. This may be because the learned features are not representative enough for quartz-3.

Table 2. Evaluation of the reconstruction accuracy for the test datasets based on the SNR and the SA metric.

		HYDICE Urban		**quartz-3**		**quartz-4**	
model	C_{R}	SA in °	SNR in dB	SA in °	SNR in dB	SA in °	SNR in dB
1D-CAE	4	**0.77**	**103.39**	**1.64**	**93.07**	**1.07**	**89.91**
DAE		0.87	102.70	2.52	90.50	1.35	88.17
NLPCA		1.00	101.03	2.34	90.65	1.39	88.07
1D-CAE	8	**0.82**	**103.27**	**2.07**	**91.66**	**1.25**	**88.57**
DAE		0.92	101.89	2.59	90.29	1.38	88.07
NLPCA		1.01	100.97	2.39	89.61	1.39	88.04
1D-CAE	16	**0.95**	**101.77**	2.65	87.98	**1.34**	**88.26**
DAE		1.19	99.70	2.63	**90.04**	1.39	88.04
NLPCA		1.23	98.58	**2.42**	88.63	1.40	87.98

The evaluation results of the different spectral unmixing methods are shown in Figure 2 for quartz-4, Figure 3 for quartz-3, and Figure 4 for HYDICE Urban. The original uncompressed data are used as reference and are denoted as $C_{\mathrm{R}} = 1$ in the figure legends. It is important to mention that the scaling of the ϵ_{RMSE} was chosen to make the tiny differences between the compression methods visible. Figure 2 shows the spectral unmixing results of the FCLS-AD and ELMM-AD methods for dataset quartz-4, where the ELMM-AD method achieves slightly better results. For $C_{\mathrm{R}} = 4$ and $C_{\mathrm{R}} = 8$, the 1D-CAE method performs best, and for $C_{\mathrm{R}} = 16$, the DAE method performs best. However, the differences in the results occur only on the third and fourth decimal places and are negligible. The same observation applies to the results in Figure 3, where the aforementioned smoothing effect also occurs. For example, the unmixing error ϵ_{RMSE} in the FCLS-AD for $C_{\mathrm{R}} = 4$ and $C_{\mathrm{R}} = 8$ of the DAE is smaller than in the baseline. The smoothed spectral signature unintentionally improves unmixing. Since the differences between the different compression methods and compression rates turn out to be small, the effect of transferability to unknown datasets cannot be conclusively assessed. Figure 4 shows the results for the HYDICE Urban dataset, where LQM achieves a higher error than the ELMM-AD, although the parameters for the transformation into the albedo domain are not known. The results of the individual unmixing methods for the different compression methods and compression rates are almost identical. The spectral unmixing results show that no significant differences between the different compression methods and compression rates for all three datasets exist.

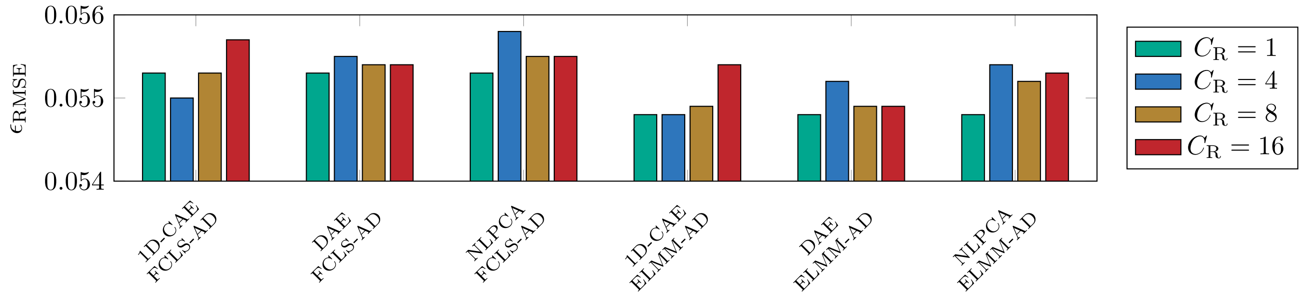

Figure 2. Spectral unmixing results of the quartz-4 dataset.

6. CONCLUSION AND OUTLOOK

We evaluated the information loss of the 1D-CAE for $C_{\mathrm{R}} = 4$, $C_{\mathrm{R}} = 8$, and $C_{\mathrm{R}} = 16$ with the SNR and SA metric as well as spectral unmixing application. Since the training and test datasets contain only a small amount of samples for machine learning based compression methods, the 1D-CAE achieved high reconstruction accuracy. The 1D-CAE outperforms the comparison methods except for the $C_{\mathrm{R}} = 16$ case on the quartz-3 dataset. Since the SA value is below 3°, the 1D-CAE approach produces sufficiently high quality in the reconstruction of the spectral signatures.

The spectral unmixing results show, that there is little difference in the error measure between the baseline and compression rates $C_{\mathrm{R}} = 4$, $C_{\mathrm{R}} = 8$, and $C_{\mathrm{R}} = 16$ of the 1D-CAE procedure, and this is true for all three

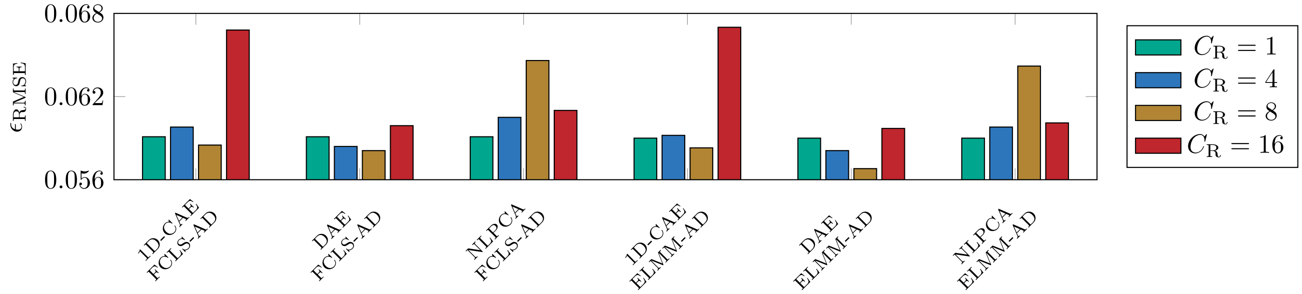

Figure 3. Spectral unmixing results of the quartz-3 dataset. The training of the compression networks has been done with the quartz-4 dataset.

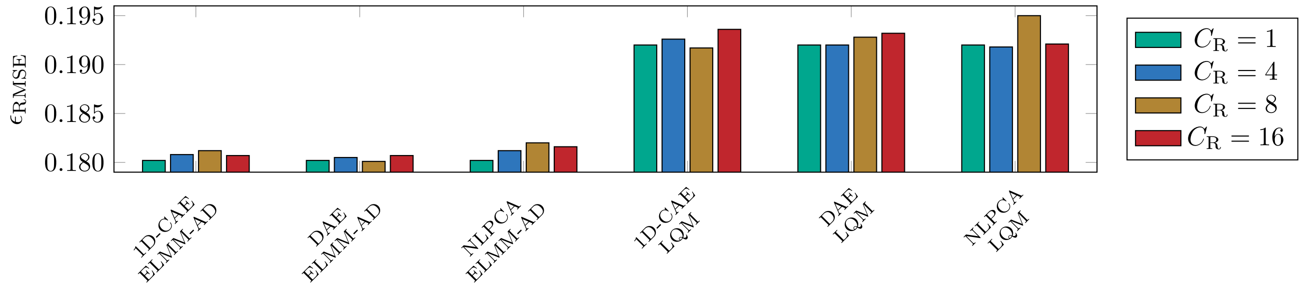

Figure 4. Spectral unmixing results of the HYDICE Urban dataset.

datasets. The difference between the 1D-CAE and the DAE, as well as the state-of-the-art NLPCA method, is small. No direct correlation between the SA and SNR results with the unmixing results could be identified. Spectral unmixing does not respond to reconstruction errors as strongly as the SA metric does. As a result, the reconstruction accuracy of the 1D-CAE for small hyperspectral datasets is high and achieves useable performance for the spectral unmixing task. Considering only the application of spectral unmixing, the user can choose a compression rate of $C_{\mathrm{R}} = 16$ without significantly worsening the results.

In further investigations, we plan to increase the compression rates and evaluate the information loss based on spectral unmixing. Furthermore, it would be of great interest how the results change when using more complex datasets that have a higher number of training and test data and different materials. As the proposed method only considers compression of the spectral dimension, spatial compression can be included to increase the compression rate. A 2D-Autoencoder could be applied for the additional spatial compression.

REFERENCES

[1] Bürsing, H. and Gross, W., "Hyperspectral imaging: future applications in security systems," *Advanced Optical Technologies* **6**(2), 61–66 (2017).

[2] Govender, M., Chetty, K., and Bulcock, H., "A review of hyperspectral remote sensing and its application in vegetation and water resource studies," *Water Sa* **33**(2), 145–151 (2007).

[3] Nasrabadi, N. M., "Hyperspectral target detection : An overview of current and future challenges," *IEEE Signal Processing Magazine* **31**(1), 34–44 (2014).

[4] Keshava, N. and Mustard, J., "Spectral unmixing," *IEEE Signal Processing Magazine* **19**(1), 44–57 (2002).

[5] Qian, S.-E., Hollinger, A. B., Williams, D., and Manak, D., "Vector quantization using spectral index-based multiple subcodebooks for hyperspectral data compression," *IEEE Transactions on Geoscience and Remote Sensing* **38**(3), 1183–1190 (2000).

[6] Ryan, M. J. and Arnold, J. F., "Lossy compression of hyperspectral data using vector quantization," *Remote Sensing of Environment* **61**(3), 419–436 (1997).

[7] Du, Q., Zhu, W., and Fowler, J. E., "Implementation of low-complexity principal component analysis for remotely sensed hyperspectral-image compression," in [*2007 IEEE Workshop on Signal Processing Systems*], 307–312 (2007).

[8] Lim, S., Sohn, K. H., and Lee, C., "Principal component analysis for compression of hyperspectral images," in [*IGARSS 2001. Scanning the Present and Resolving the Future. Proceedings. IEEE 2001 International Geoscience and Remote Sensing Symposium (Cat. No.01CH37217)*], **1**, 97–99 vol.1 (2001).

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

[10] Abousleman, G. P., Marcellin, M. W., and Hunt, B. R., "Compression of hyperspectral imagery using the 3-d dct and hybrid dpcm/dct," *IEEE transactions on geoscience and remote sensing* **33**(1), 26–34 (1995).

[11] Pickering, M. R. and Ryan, M. J., "Compression of hyperspectral data using vector quantisation and the discrete cosine transform," in [*Proceedings 2000 International Conference on Image Processing (Cat. No. 00CH37101)*], **2**, 195–198, IEEE (2000).

[12] Du, Q. and Fowler, J. E., "Hyperspectral image compression using jpeg2000 and principal component analysis," *IEEE Geoscience and Remote sensing letters* **4**(2), 201–205 (2007).

[13] Penna, B., Tillo, T., Magli, E., and Olmo, G., "Transform coding techniques for lossy hyperspectral data compression," *IEEE Transactions on Geoscience and Remote Sensing* **45**(5), 1408–1421 (2007).

[14] Ballé, J., Laparra, V., and Simoncelli, E. P., "End-to-end optimized image compression," *arXiv preprint arXiv:1611.01704* (2016).

[15] Theis, L., Shi, W., Cunningham, A., and Huszár, F., "Lossy image compression with compressive autoencoders," *arXiv preprint arXiv:1703.00395* (2017).

[16] Cheng, Z., Sun, H., Takeuchi, M., and Katto, J., "Deep convolutional autoencoder-based lossy image compression," in [*2018 Picture Coding Symposium (PCS)*], 253–257, IEEE (2018).

[17] Wadströmer, N. and Gustafsson, D., "Non-linear hyperspectral subspace mapping using stacked autoencoder," in [*The 29th Annual Workshop of the Swedish Artificial Intelligence Society (SAIS) 2–3 June 2016, Malmö, Sweden*], 5 (2016).

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

[19] Myasnikov, E., "Hyperspectral data dimensionality reduction using nonlinear autoencoders," in [*CEUR Workshop Proceedings*], 33–36 (2020).

[20] Licciardi, G. and Chanussot, J., "Spectral transformation based on nonlinear principal component analysis for dimensionality reduction of hyperspectral images," *European Journal of Remote Sensing* **51**(1), 375–390 (2018).

[21] Priya, S., Ghosh, R., and Bhattacharya, B., "Non-linear autoencoder based algorithm for dimensionality reduction of airborne hyperspectral data," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* **42**, 593–598 (2019).

[22] Minkin, A., "The application of autoencoders for hyperspectral data compression," in [*2021 International Conference on Information Technology and Nanotechnology (ITNT)*], 1–4, IEEE (2021).

[23] Dua, Y., Singh, R. S., Parwani, K., Lunagariya, S., and Kumar, V., "Convolution neural network based lossy compression of hyperspectral images," *Signal Processing: Image Communication* **95**, 116255 (2021).

[24] Kuester, J., Gross, W., and Middelmann, W., "1D-Convolutional Autoencoder Based Hyperspectral Data Compression," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* **43**, 15–21 (2021).

[25] Keshava, N. and Mustard, J. F., "Spectral unmixing," *IEEE signal processing magazine* **19**(1), 44–57 (2002).

[26] Ioffe, S. and Szegedy, C., "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *CoRR* **abs/1502.03167** (2015).

[27] Dobigeon, N., Tourneret, J.-Y., Richard, C., Bermudez, J. C. M., McLaughlin, S., and Hero, A. O., "Nonlinear unmixing of hyperspectral images: Models and algorithms," *IEEE Signal Processing Magazine* **31**(1), 82–94 (2014).

[28] Clark, R. N. and Roush, T. L., "Reflectance spectroscopy: Quantitative analysis techniques for remote sensing applications," *Journal of Geophysical Research: Solid Earth* **89**(B7), 6329–6340 (1984).

[29] Dobigeon, N., Altmann, Y., Brun, N., and Moussaoui, S., "Linear and nonlinear unmixing in hyperspectral imaging," in [*Data Handling in Science and Technology*], Ruckebusch, C., ed., **30**, 185–224, Elsevier (2016).

[30] Meganem, I., Déliot, P., Briottet, X., Deville, Y., and Hosseini, S., "Linear–quadratic mixing model for reflectances in urban environments," *IEEE Transactions on Geoscience and Remote Sensing* **52**(1), 544–558 (2013).

[31] Hapke, B., [*Theory of reflectance and emittance spectroscopy*], Cambridge University Press (1993).

[32] Nascimento, J. M. and Bioucas-Dias, J. M., "Unmixing hyperspectral intimate mixtures," in [*Image and signal processing for remote sensing XVI*], **7830**, 78300C, International Society for Optics and Photonics (2010).

[33] Close, R., Gader, P., Wilson, J., and Zare, A., "Using physics-based macroscopic and microscopic mixture models for hyperspectral pixel unmixing," in [*Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XVIII*], Shen, S. S. and Lewis, P. E., eds., **8390**, 469 – 481, International Society for Optics and Photonics, SPIE (2012).

[34] Heylen, R. and Gader, P., "Nonlinear spectral unmixing with a linear mixture of intimate mixtures model," *IEEE Geoscience and Remote Sensing Letters* **11**(7), 1195–1199 (2013).

[35] Heinz, D., Chang, C.-I., and Althouse, M. L., "Fully constrained least-squares based linear unmixing," in [*IEEE 1999 International Geoscience and Remote Sensing Symposium. IGARSS'99 (Cat. No. 99CH36293)*], **2**, 1401–1403, IEEE (1999).

[36] Smith, M. O., Adams, J. B., and Sabol, D. E., "Spectral mixture analysis-new strategies for the analysis of multispectral data," in [*Imaging spectrometry – a tool for environmental observations*], 125–143, Springer (1994).

[37] Borsoi, R. A., Imbiriba, T., Bermudez, J. C. M., Richard, C., Chanussot, J., Drumetz, L., Tourneret, J.-Y., Zare, A., and Jutten, C., "Spectral variability in hyperspectral data unmixing: A comprehensive review," *arXiv preprint arXiv:2001.07307* (2020).

[38] Griffin, M. K. and Burke, H.-h. K., "Compensation of hyperspectral data for atmospheric effects," *Lincoln Laboratory Journal* **14**(1), 29–54 (2003).

[39] Drumetz, L., Veganzones, M.-A., Henrot, S., Phlypo, R., Chanussot, J., and Jutten, C., "Blind hyperspectral unmixing using an extended linear mixing model to address spectral variability," *IEEE Transactions on Image Processing* **25**(8), 3890–3905 (2016).

[40] Anastasiadis, J. and Heizmann, M., "CNN-based augmentation strategy for spectral unmixing datasets considering spectral variability," in [*Image and Signal Processing for Remote Sensing XXVI*], **11533**, 188 – 199, SPIE (2020).

[41] (U.S.), G. R. L., "HyperCube Sample Data Set: HYDICE sensor imagery URBAN." ERDC Knowledge Core, https://hdl.handle.net/11681/2925. (Accessed: 28 July 2022).

[42] Center, T. A. G., "Hypercube." U.S. Army Corps of Engineers, https://www.agc.army.mil/What-we-do/Hypercube/. (Accessed: 28 July 2022).

[43] Jain, A. K., [*Fundamentals of digital image processing*], Prentice-Hall, Inc. (1989).

[44] Kruse, F. A., Lefkoff, A., Boardman, J., Heidebrecht, K., Shapiro, A., Barloon, P., and Goetz, A., "The spectral image processing system (sips)—interactive visualization and analysis of imaging spectrometer data," *Remote sensing of environment* **44**(2-3), 145–163 (1993).

[45] Kingma, D. and Ba, J., "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980* (2014).

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