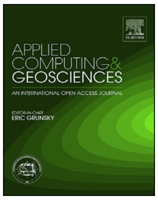
Contents lists available at ScienceDirect

Applied Computing and Geosciences

journal homepage: www.journals.elsevier.com/applied-computing-and-geosciences



Robust input layer for neural networks for hyperspectral classification of data with missing bands

Laurent Fasnacht, Philippe Renard, Philip Brunner [*]

Université de Neuchâtel, Centre d'Hydrogéologie et de Géothermie, Rue Emile Argand 11, CH-2000 Neuchâtel, Switzerland

ABSTRACT

Hyperspectral classification using artificial neural networks is commonly applied on camera dependent interpolated data, or on the results of a dimensionality reduction algorithm. While these methods usually produce satisfactory results, they have severe limitations when part of the spectrum is missing, for example when parts of the image are overexposed or affected by bad pixels. This article presents an input layer based on the Haar transform for artificial neural networks used for hyperspectral data classification. This input layer is designed to perform efficiently with incomplete data and is independent of the specific bands used by the camera. This could enable providing pre-trained neural networks, which can be used with a camera with different specifications than the one used for training. This paper shows that a classifier for mineral identification built using this approach performs better than standard normalization on incomplete spectra, and similarly on complete spectra. Additionally, it shows that such a classifier matches local spectral features, and therefore that the artificial neural network is matching the spectrum shape.

1. Introduction

Artificial neural networks (ANN) have been widely used for various classification problems, including hyperspectral data classification. The flexibility and the performance of ANNs make them common for numerous applications such as classifying aerial images (Ratle et al., 2010; Makantasis et al., 2015; Hu et al., 2015; LiangQi, 2016), food industry (Gamal ElMasry et al., 2009), bacteria identification (Goodacre et al., 1998), nitrogen concentration estimation in rice leaves (Yi et al., 2007). In addition to classification, neural networks can also be used for unmixing (Plaza et al., 2009).

Hyperspectral images have many bands. The number of these bands and their center wavelengths depend on the camera model. Usually, there is a strong correlation between bands, as the spectral features are commonly wide enough to span over multiple bands. The papers previously cited rely on dimensionality reduction before the ANN. This has the advantage of reducing the size of the input data (thus reducing the requirements of the neural network) and of separating the input features. However, dimensionality reduction has some drawbacks:

- The number of dimensions kept is a choice to be made, which adds additional hyperparameters.
- Dimensionality reduction is not likely to isolate spectral features. This means that one coefficient of the output depends on the whole input spectrum.
- Dimensionality reduction techniques consist in finding the transform that can express the diversity of data using fewer variables. As such, a dimensionality reduction is dependent on the data for which it was computed and its parameters can be seen as additional hyperparameters of the network. Therefore, the network will have reduced efficiency if additional training or retraining is applied using different data than those that were used to compute the transform.

Due to the complex acquisition procedure of hyperspectral data, the data acquired is of imperfect quality. Neural networks are resilient to noise, but missing data is an issue. Missing data can arise for three different reasons (see Fig. 1):

- The sensor has bad pixels, which should be masked.
- The sensor is saturated for some bands, either for the sample measurement or for the white reference. As the value of such bands is capped to the maximum possible output value of the sensor, these bands should be masked as the data is not reliable.
- Since the training is computationally demanding, pre-trained neural networks can be distributed. In that case, it can happen that the bands of the data are not the same as those that were used during the creation of the ANN, resulting in missing bands.

Handling data with missing bands is challenging in general. Imputation is commonly used to compute the missing values, usually using

[*] Corresponding author.
E-mail address: philip.brunner@unine.ch (P. Brunner).

https://doi.org/10.1016/j.acags.2020.100034
Received 14 May 2020; Received in revised form 2 August 2020; Accepted 2 August 2020
Available online 21 August 2020

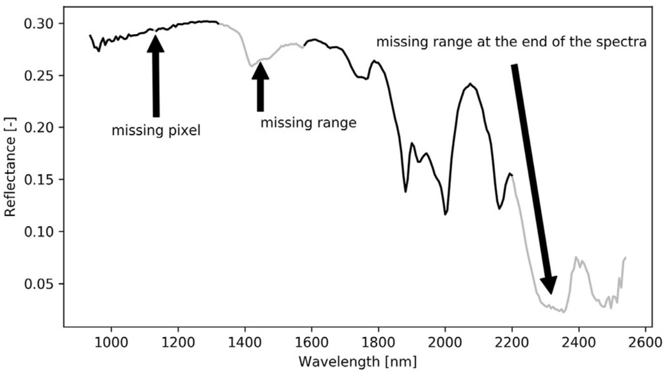


Fig. 1. Synthetic example of missing spectral data.

interpolation, replacement by the average of the data, or replacement by the maximum likelihood (Tresp et al., 1994). Interpolation is however not straightforward for missing spectral ranges, can be computationally demanding, and may lead to providing bad quality input data to the ANN. The other approach available is marginalization, which consists in ignoring bad values. It cannot be directly used with ANNs as they require a complete input layer. Imputation and marginalization can be used both with dimensionality reduction (Dray and Josse, 2015) or for classification algorithms (Wagstaff et al., 2004), at the cost of added algorithmic complexity.

In this article, we present an input preprocessing method based on the Haar wavelet decomposition. This method is designed to be independent of the content of the input data set, resilient to missing data, and able to cope with different image specifications (bands center and spectral range).

The impact on the learning rate and the accuracy of the method is compared to standard normalization and is evaluated using a dataset of 73 different minerals. Minerals in this dataset were selected to represent all the mineral classes and the most common mineral occurrences. The spectra originate from rock samples, of well characterized, but not necessarily pure minerals. This ensure the diversity required to correcly train and test a classifier. This dataset is presented in more detail in (Fasnacht et al., 2019).

This paper considers only single spectrum classification, as applying spectral-spatial classification (using convolutional neural networks) would "smooth" the difference between the various methods compared.

2. Method

2.1. Concepts and notations

ANNs are inspired by biology, but practically they consist in a composition of linear operators and simple non-linear operators (activation function). The standard neural network is a sequence of layers, each of which is a composition of a linear operator with unknown coefficients (parameters) and an activation function. These layers link an input (source data) to an output (the prediction), and can be visualized as a graph (Fig. 2). The parameters of all the operators of an ANN have to be determined during a training phase. This training phase requires a lot of input data associated with known output (ground truth) and consists in finding the coefficients which create the optimal correspondence between the output of the neural network and the ground truth. The training phase is demanding both in computing power and in the quantity of data, as the operator corresponding to the neural network is highly unspecific. To reduce the training phase cost, it is possible to retrain only the last layer(s) of a neural network, if the classes have changed (partial retraining).

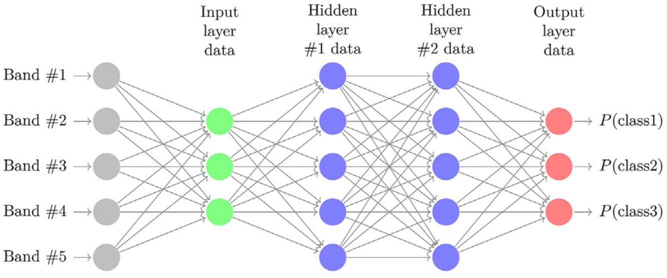


Fig. 2. Example of an artificial neural network, with two hidden layers.

To avoid overfitting a feature, usually, dropouts are applied during the training phase. This consists of randomly dropping a fraction of the coefficients in the neural network while compensating on the other coefficients (Srivastava et al., 2014). This has the advantage that the network is "forced" to learn multiple ways to identify the same data. Dropouts can technically also be applied at the input layer.

Once the training is done, applying the neural network to the data is usually fast, as it consists of direct computations.

Throughout this document, we will work with vectors including missing values. For a given vector $\overrightarrow{v}$, we define $\overrightarrow{v}^f$ the vector where missing values are replaced by 0. We also defined $\overrightarrow{v}^m$ the mask vector, where $v_i^m = 1$ if the value v_i is missing, or 0 otherwise. Moreover, we define $\overrightarrow{v} \leq x$ to be the vector of the point-wise comparison, where each element is 1 if $v_i \leq x$ and 0 otherwise. This is a convention that is commonly used by numerical libraries.

There are some discrepancies between what terms are used to describe neural network layers depending on the context. In this document, we will use the following:

- the term *layer* is used to refer to the set of operators linking the previous layer data to the current layer data. For example, the first hidden layer is the operators linking the input layer data to the first hidden layer data.
- the term *layer data* corresponds to the actual data vector.
- as a consequence, since normalization can be seen as a part of the network, we use the term *input layer* to refer to the operators linking the raw data to the input layer data of the neural network.

2.2. The Haar inspired input layer (HIIL)

The input layer is inspired by the discrete wavelet transform, using Haar wavelets. Discrete wavelets transform consists in representing a function (in the present case, the measured reflectance) as a sum of orthonormal wavelets. The Haar wavelet, commonly used in image processing, is a piece-wise constant wavelet which makes it easy to use. There is abundant literature about Haar wavelets (for example (Radomir and Falkowski, 2003; Porwik and Lisowska, 2004)), therefore only the specific variation used in this paper will be presented.

The transformation proposed in this paper requires 4 parameters: the minimum wavelength λ_{min}, the maximum wavelength λ_{max}, the number of subdivision levels N, and the required ratio of valid values $\delta \in [0, 1]$. For example, for $\delta = 0.9$, every coefficient computed with less than 90% of the spectral range covered will be considered invalid. A missing value is considered as a 0 in the integrals.

We then proceed as follows, where r is the measured reflectance:

1 The first coefficient is the integral of the reflectance:

$$a_0 \leftarrow \int_{\lambda_{min}}^{\lambda_{max}} r d\lambda \tag{1}$$

2 If $N \geq 1$, for the first subdivision level we compute:

$$a_1 \leftarrow \int_{\lambda_{min}}^{\frac{\lambda_{min}+\lambda_{max}}{2}} r d\lambda - \int_{\frac{\lambda_{min}+\lambda_{max}}{2}}^{\lambda_{max}} r d\lambda \tag{2}$$

3 If $N \geq 2$, for the second subdivision level we compute:

$$\begin{aligned} a_2 &\leftarrow \int_{\lambda_{min}}^{\frac{\lambda_{min}+\lambda_{max}}{4}} r d\lambda - \int_{\frac{\lambda_{min}+\lambda_{max}}{4}}^{\frac{\lambda_{min}+\lambda_{max}}{2}} r d\lambda, \\ a_3 &\leftarrow \int_{\frac{\lambda_{min}+\lambda_{max}}{2}}^{3\frac{\lambda_{min}+\lambda_{max}}{4}} r d\lambda - \int_{3\frac{\lambda_{min}+\lambda_{max}}{4}}^{\lambda_{max}} r d\lambda \end{aligned} \tag{3}$$

4 Repeat this procedure until the number of subdivision levels has been reached. The sequence $a_0 \ldots a_{2^N-1}$ is the output of the transform.

An example of this transform is shown in Fig. 3.

In addition, set $a_i = 0$ for all i where less than δ of the spectral range is covered by actual measurements.

In practice, the reflectance output is a vector $\overrightarrow{r} \in \mathbb{R}^k$, where k is the number of bands of the camera. The physical characteristics of the camera determines for each band the center wavelength and the width of the band. The widths of the bands can be visualized as a vector $\overrightarrow{d} \in \mathbb{R}^k$.

This transformation is a linear operator that can be discretized in a matrix $\mathbf{M} \in \mathbb{R}^{2^N-1 \times k}$. Each row contains the coefficients required to obtain an output element a_i. The first row is constructed to cover the whole spectral range covered by the integral. Note that the end of the range can partially cover the bands, leading to coefficients $\tilde{d}_k \leq d_k$ and $\tilde{d}_l \leq d_l$. The row is structured as follows:

$$M_0 = [0 \quad \ldots \quad 0 \quad \tilde{d}_k \quad d_{k+1} \quad \ldots \quad d_{l-1} \quad \tilde{d}_l \quad 0 \quad \ldots \quad 0] \tag{4}$$

The next row, M_1, has both a positive and a negative range. As previously, the bands at the side can be partially covered by the integral. Moreover, the central term $\tilde{d}_c$ is a combination of the positive and negative integral, so it can be any value between $-d_c$ and d_c, depending on where $\frac{\lambda_{min}+\lambda_{max}}{2}$ is situated in the band. The row structure is as follows:

$$M_1 = [0 \quad \ldots \quad 0 \quad \tilde{d}_k \quad d_{k+1} \quad \ldots \quad d_{c-1} \quad \tilde{d}_c \quad -d_{c+1} \quad \ldots \quad -d_{l-1} - \tilde{d}_l \quad 0 \quad \ldots \quad 0] \tag{5}$$

The rest of the matrix is constructed likewise. A visualization can be seen in Fig. 4.

Additionally, the matrix $\mathbf{N}$ is derived from $\mathbf{M}$, in which each element corresponding to a non-zero element of $\mathbf{M}$ is the inverse of the number of non-zero elements of the row in $\mathbf{M}$. Therefore, the sum of each line in $\mathbf{N}$ is 1, and it can be expressed as follows:

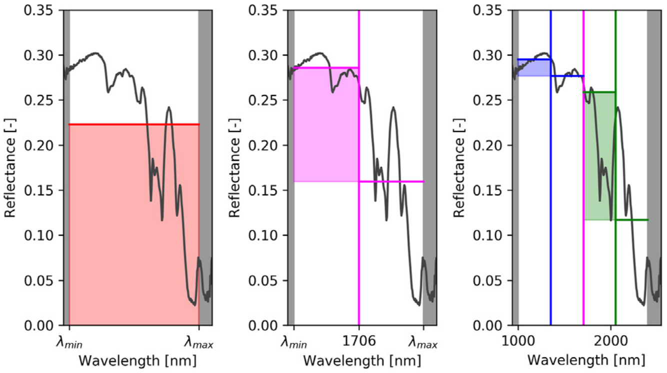


Fig. 3. Example of a Haar transform with $N = 2$ levels of subdivisions, between $\lambda_{min} = 1000$nm and $\lambda_{max} = 2400$nm. The coefficients are $a_0 = 0.22 \cdot 1400$ (red area), (pink area), $a_2 = 0.02 \cdot 350$ (blue area), $a_3 = 0.14 \cdot 350$ (green area).

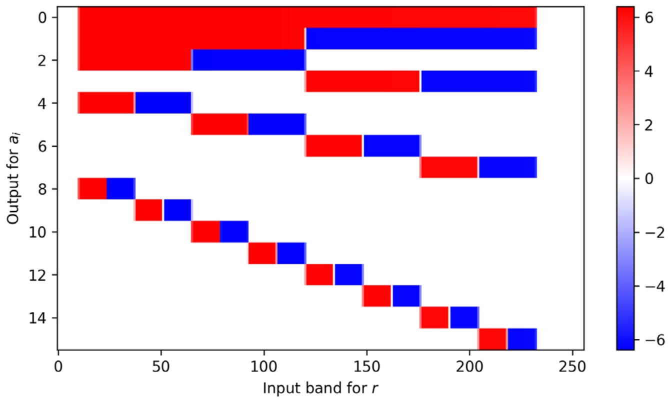


Fig. 4. Example of matrix M with $N = 4$. The values on the side and in the center are clearly smaller due to limits of the integral not aligning on a band limit. Due to the relatively small difference in the colors, it is not possible to see that the band widths span between 6.2nm and 6.4nm.

$$N_{i,j} = \begin{cases} 0 & \text{if} \quad M_{i,j} = 0 \\ \dfrac{1}{\#\{M_{i,k} \neq 0 \forall k\}} & \text{otherwise.} \end{cases} \tag{6}$$

For a given reflectance vector $\overrightarrow{r}$ with missing values, we can then do the decomposition as follows:

$$\overrightarrow{a} = \left(\mathbf{M} \cdot \overrightarrow{r}^f\right) \cdot (\mathbf{N} \cdot \overrightarrow{r}^m \leq (1-\delta)) \tag{7}$$

2.3. Evaluation of the efficiency of the Haar-inspired input layer

To test that HIIL is an efficient input layer, the following points need to be considered:

- the training of the network is efficient
- the accuracy of the network is good
- the network is robust to missing data in the input

To evaluate this, a dataset consisting of 3′842′482 spectra of 73 different classes was used (Fig. 5). Of these points, 974′544 have at least one missing band, due to bad pixels of the sensor, or due to saturation.

The dataset was filtered to remove the incomplete spectra, and was split into three non-overlapping parts:

- the training dataset, containing 85% of the data points,
- the validation dataset, containing 10% of the data points,
- the test dataset, containing 5% of the data points.

Additionally, a second test dataset was created, containing 5% of the points of the original (non-filtered) dataset, of which 30% of the spectral data had been additionaly erased, by randomly selecting 30% of the non-missing bands. Therefore, each spectra in this set simultaneously exhibit

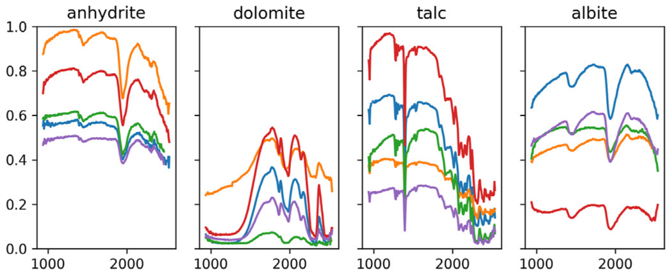


Fig. 5. Example reflectance spectra from the dataset, for 4 different classes. For each class, 5 random spectra were selected.

missing data both due to sensor defects and from the random removal, and is therefore very likely to have small spectral ranges missing. While this dataset contain some spectra that were the training and validation, overfitting won’t occur, due to the large amount of spectral data removed.

For all these steps, we used a neural network consisting of three dense hidden layers, each consisting of a linear operator, a bias, and a ReLU, of sizes 1024, 1024, 256. This is a commonly used setup (LeCun et al., 2015).

Having a too small ANN size can constrain the learning, as there would not be enough coefficients to fully model the relation between the input and the output. Therefore, the size of the ANN was chosen to be large enough, as the goal is the evaluation of the influence of the input layer. In practice, a smaller network size should be chosen in order to increase computational efficiency.

The training was performed using an Adam optimizer (KingmaJimmy, 2014). The training data was balanced between classes, and the accuracy of the classifier on the validation dataset was stored after each epoch.

As input layers, we used:

- The identity operator, which consists in using directly the raw data as the input layer. This is known to create optimization problems during the training phase but is useful as a reference point.
- The standard normalization. For a given reflectance data r, we use as an input layer: $\frac{r - mean(r)}{std(r)}$.
- HIIL, with two different transform depths $N = 6$ and $N = 8$

To avoid over-fitting, dropouts were added at the input layer and each hidden layer.

For each of the input layers, we evaluated the accuracy after the classifier after 1′000 epochs, for various drop-out parameters.

The goal of HIIL is to be used on data with missing values, so it needs to be evaluated on the second test dataset. However, since the other input layers require a full spectrum, we evaluated them with the spectrum linearly interpolated. This is not an optimal approach, but the more advanced approaches would require statistics about the input data, which may not be available. Moreover, this approach is computationally demanding. We also evaluated HIIL in these conditions, to see how interpolated values impact the accuracy of the classifier.

The classifier using the HIIL input layer was also evaluated on partial spectra. The principle consists in choosing a center band, a width of the interval, and to mark every other band as invalid. The accuracy of the prediction can then be computed. This can be done for every interval and every class.

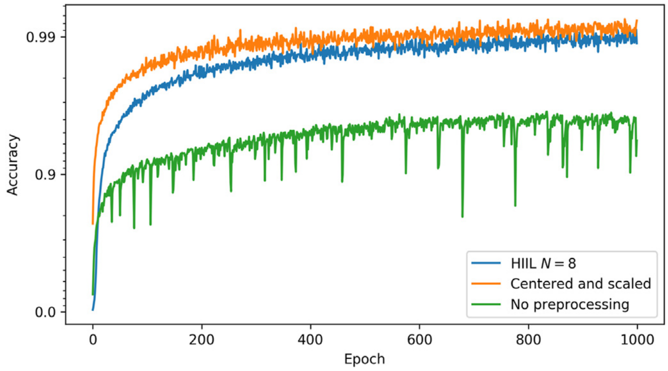


Fig. 6. Accuracy of the ANN on the validation set. The input layer dropout is disabled, and the dropout after each layer was set to 0.25.

3. Results

To assess the efficiency of the training, we compared the evolution of the accuracy on the validation dataset throughout learning (Fig. 6). We can observe that the identity input layer limits accuracy (it reaches a plateau), but other input layers yield good results, even if HIIL has a slower learning rate.

By using the first test set for each input layer while varying the dropout, we can observe that the results are similar between the HIIL and the centered and scaled normalization (Table 1). The dropout after the input layer has an important negative effect on the training, but HIIL is less sensitive to it.

As we can see in Table 2, the accuracy using the HIIL input layer is better than when using other input layers, even with the invalid values replaced by 0.

For partial spectra, the results for calcite, gypsum, and quartz can be seen in Figs. 7–9 respectively. Observe that classification accuracy is good as soon as the specific spectral features of the mineral are given to the classifier. For classes with nearly no specific spectral features like quartz (Fig. 9), nearly the full spectrum is required in order to classify the spectrum correctly. Intuitively, this is due to the classifier having to ensure that each feature of the training dataset is absent.

The impact of the HIIL input layer on computation requirements to train and use the neural network was negligible.

4. Discussion & conclusions

Complete spectral data was classified correctly 98% of the time. Spectral data with 30% of the data removed was still classified correctly 70% of the time. We can observe that the input layer dropout increases the accuracy of the classifier. We also showed that a classifier built using the HIIL has the ability to find and match local features.

As expected, the presented input layer has low computational requirements, therefore it can be easily applied to practical problems. An unexpected result is that HIIL performs better if the data is linearly interpolated to fill holes.

HIIL is focusing on local features instead of global ones, unlike the methods relying on dimensionality reduction. This creates a clear relationship between specific spectral regions and neuron activation, which is more intuitive than to have activation based on a linear transform of the whole spectrum. Compared to the classical normalization approach, we showed that it performs similarly if the data contains no missing data, and performs better with real data with missing bands. It is also much simpler to implement than alternative approaches.

The high classification accuracy using HIIL is due to the following reasons:

- The HIIL’s output is 0-centered, which is compatible with the classical assumptions of ANNs layers (activation functions and dropouts)
- Missing input data generates output similar to dropout at the input layer. Therefore the network is in similar conditions during the training and prediction phases.

Table 1
Comparison of the accuracy on the first test set using various input layers, with layer dropout 0.00, 0.25, 0.50.

Input layer	Input dropout = 0.0	Input dropout = 0.25	Input dropout = 0.50
Identity	97.0%, 94.5%, 95.5%	53.0%, 35.6%, 37.6%	6.0%, 16.2%, 20.3%
Centered and scaled	**99.0%, 99.2%, 98.9%**	95.3%, 94.3%, 93.3%	64.7%, 80.9%, 87.1%
HIIL ($N = 6$)	98.6%, 98.6%, 98.0%	97.2%, 96.5%, 95.4%	90.2%, 87.5%, 86.8%
HIIL ($N = 8$)	98.9%, 99.0%, 98.4%	**98.4%, 97.6%, 96.7%**	**93.2%, 92.3%, 91.6%**

Table 2
Comparison of the accuracy on the second test set (with missing values) of the various input layers, with layer dropout 0.00, 0.25, 0.50.

Input layer	Input dropout = 0.0	Input dropout = 0.25	Input dropout = 0.50
Identity (interpolated)	61.6%, 64.7%, 67.5%	31.9%, 28.0%, 31.6%	5.0%, 14.2%, 17.1%
Centered and scaled (interpolated)	59.2%, 62.9%, 62.0%	57.5%, 57.7%, 58.0%	35.2%, 49.7%, 53.0%
HIIL (N = 6, interpolated)	57.2%, 60.1%, 61.5%	75.4%, 77.3%, 74.7%	68.9%, 67.2%, 66.7%
HIIL (N = 8, interpolated)	59.6%, 63.9%, 62.9%	74.6%, 75.5%, 73.9%	71.9%, 72.2%, 71.9%
HIIL (N = 6)	13.4%, 16.9%, 15.6%	64.8%, 65.5%, 64.0%	70.0%, 69.0%, 67.2%
HIIL (N = 8)	16.2%, 18.4%, 19.0%	66.1%, 67.1%, 65.2%	69.2%, 70.4%, 69.6%

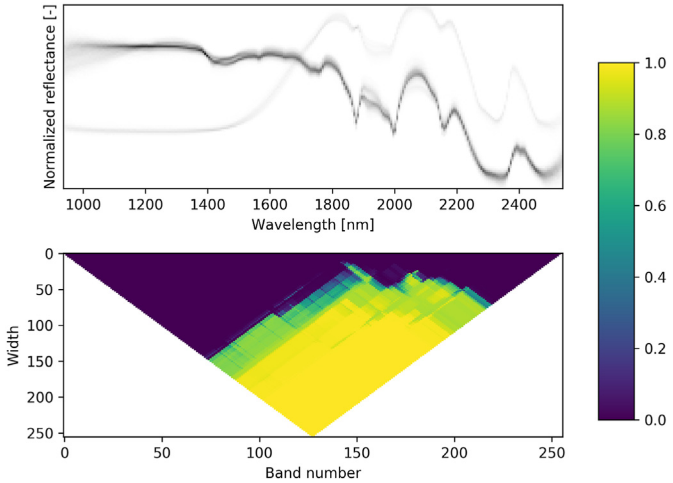


Fig. 7. Accuracy of classification of calcite spectra, depending on the spectral area which is provided to the ANN. For example, we can see that with an interval of 50 bands around band 150, the accuracy of the classifier is around 80%. Observe that as soon as the wavelengths from the farthest part of the spectra are covered, we get good recognition. This makes sense since this is the part of the spectrum in which there are variations.

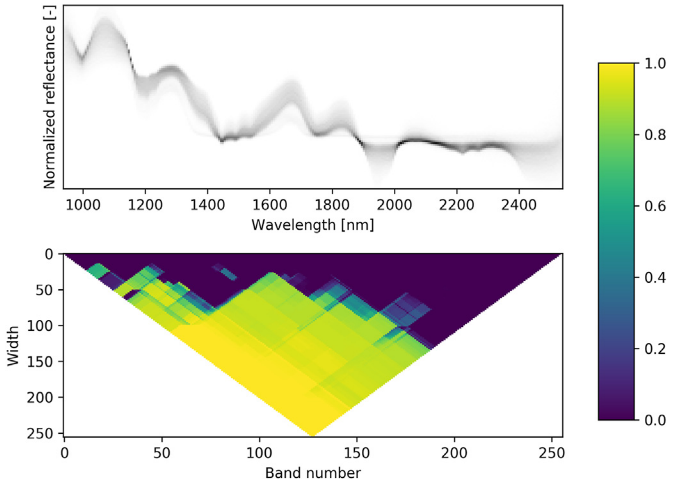


Fig. 8. Accuracy of classification of gypsum spectra, depending on the spectral area which is provided to the ANN. Similarly to Fig. 7, we can observe that the relevant part of the spectrum is the lowest wavelengths. This matches our intuition since it's the part of the spectrum with distinctive shape.

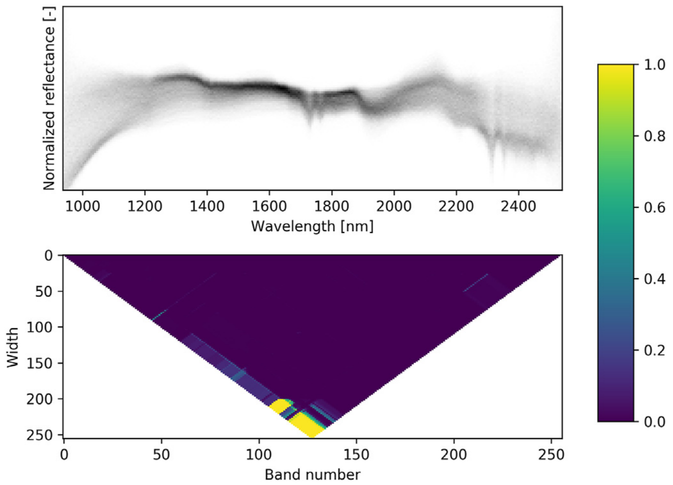


Fig. 9. Accuracy of classification of quartz spectra, depending on the spectral area which is provided to the ANN. As quartz has nearly no spectral variations in the wavelengths considered, nearly all the spectrum is required to have a correct classification. This can be intuitively understood as matching the absence of features in the whole spectrum.

- The input layer data scale is proportional to the spectral range covered by the integrals. This tends to naturally increase the influence of large spectral features compared to small ones (which could be noise), therefore making the ANN focus on general tendency instead of noise.
- The transform is independent of the input data, as it is applied individually on each spectrum.

Finally, the low computational impact of the HIIL is due to the fact that its linear nature is compatible with the kind of operators used in ANNs, and therefore benefits from the optimization usually applied in the ANN software libraries to speed up computations.

The increase of accuracy when HIIL is applied with linearly interpolated data suggests that some information is not captured by the neural network. This suggests that tuning would be required, for example by changing the ANN parameters, or by changing δ.

One limitation of this study is that it does not consider the impact of using this input layer on the size and the complexity required of the ANN. It was considered not necessary because the computational requirement was in any case quite low. Moreover, dimensionality reduction techniques were not evaluated, as numerous variant exists. Besides, the algorithmic complexity of these methods is higher and the ANN requirements are quite different. We also did not consider the re-training abilities of classifiers using the HIIL, although we expect them to be good.

We have also only considered laboratory hyperspectral measurements in our tests. Further work is required to assess the robustness of such an approach when combined with varying athmospheric conditions.

Further studies could be done on using different types of wavelets, or different transform specifications. One should also compare how this input layer performs compared to the other approaches when the camera used to train the network is not the same as the one used for prediction. Comparing its efficiency using a classifier that also uses spatial features, such as a convolutional neural network, might also provide interesting results.

HIIL is likely to be also applicable to other types of continuous data with missing parts.

To conclude, a classifier for hyperspectral mineral identification built using the Haar Inspired Input Layer has better accuracy than a classifier using standard normalization technique on incomplete spectra, and similar accuracy on complete spectra. As it has lower computational requirements, it should be considered for hyperspectral classification using artificial neural networks.

Contributions

Laurent Fasnacht: captured the data, processed the data, developed the computer software, analyzed the results, and drafted the manuscript. Philip Brunner and Philippe Renard: supervised the research and revised the manuscript.

Computer code availability

Although this article presents an algorithm (not an implementation), an example implementation based on Keras is available at https://github.com/UniNE-CHYN/HIIL-keras.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Dray, Stéphane, Josse, Julie, 2015. Principal component analysis with missing values: a comparative survey of methods. Plant Ecol. 216 (5), 657–667.

Fasnacht, Laurent, Vogt, Marie-Louise, Renard, Philippe, Brunner, Philip, 2019. A 2d hyperspectral library of mineral reflectance, from 900 to 2500 nm. Scientific data 6 (1), 1–7.

Gamal ElMasry, Wang, Ning, Vigneault, Clément, 2009. Detecting chilling injury in red delicious apple using hyperspectral imaging and neural networks. Postharvest Biol. Technol. 52 (1), 1–8.

Goodacre, Royston, Burton, Rebecca, Kaderbhai, Naheed, Woodward, Andrew M., Kell, Douglas B., Rooney, Paul J., et al., 1998. Rapid identification of urinary tract infection bacteria using hyperspectral whole-organism fingerprinting and artificial neural networks. Microbiology 144 (5), 1157–1170.

Hu, Wei, Huang, Yangyu, Li, Wei, Zhang, Fan, Li, Hengchao, 2015. Deep convolutional neural networks for hyperspectral image classification. Journal of Sensors 2015.

Kingma, Diederik P., Jimmy, Ba, 2014. Adam: A Method for Stochastic Optimization arXiv preprint arXiv:1412.6980.

LeCun, Yann, Bengio, Yoshua, Hinton, Geoffrey, 2015. Deep learning. Nature 521 (7553), 436.

Liang, Heming, Qi, Li, 2016. Hyperspectral imagery classification using sparse representations of convolutional neural network features. Rem. Sens. 8 (2), 99.

Makantasis, Konstantinos, Karantzalos, Konstantinos, Doulamis, Anastasios, Doulamis, Nikolaos, 2015. Deep supervised learning for hyperspectral data classification through convolutional neural networks. In: Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International. IEEE, pp. 4959–4962.

Plaza, Antonio, Atli Benediktsson, Jon, Boardman, Joseph W., Brazile, Jason, Bruzzone, Lorenzo, Camps-Valls, Gustavo, Chanussot, Jocelyn, Fauvel, Mathieu, Gamba, Paolo, Gualtieri, Anthony, et al., 2009. Recent advances in techniques for hyperspectral image processing. Remote Sens. Environ. 113, S110–S122.

Porwik, Piotr, Lisowska, Agnieszka, 2004. The haar-wavelet transform in digital image processing: its status and achievements. Mach. Graph. Vis. 13 (1/2), 79–98.

Radomir, S Stanković, Falkowski, Bogdan J., 2003. The haar wavelet transform: its status and achievements. Comput. Electr. Eng. 29 (1), 25–44.

Ratle, Frédéric, Camps-Valls, Gustavo, Weston, Jason, 2010. Semisupervised neural networks for efficient hyperspectral image classification. IEEE Trans. Geosci. Rem. Sens. 48 (5), 2271–2282.

Srivastava, Nitish, Hinton, Geoffrey, Krizhevsky, Alex, Sutskever, Ilya, Salakhutdinov, Ruslan, 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15 (1), 1929–1958.

Tresp, Volker, Ahmad, Subutai, Neuneier, Ralph, 1994. Training neural networks with deficient data. In: Advances in Neural Information Processing Systems, pp. 128–135.

Wagstaff, Kiri, 2004. Clustering with missing values: No imputation required. In: Banks, David, McMorris, Frederick R., Arabie, Phipps, Gaul, Wolfgang (Eds.), Classification, Clustering, and Data Mining Applications. Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN 978-3-642-17103-1, pp. 649–658.

Yi, Qiu-Xiang, Huang, Jing-Feng, Wang, Fu-Min, Wang, Xiu-Zhen, Liu, Zhan-Yu, 2007. Monitoring rice nitrogen status using hyperspectral reflectance and artificial neural network. Environ. Sci. Technol. 41 (19), 6770–6775.