CSSNET: A Learning Algorithm for the Segmentation of Compressed Hyperspectral Images
Maud Biquard, Simon Lacroix, Antoine Rouxel, Hervé Carfantan, Antoine Monmayrant, Henri Camon

To cite this version:

HAL Id: hal-03746594
https://hal.laas.fr/hal-03746594
Submitted on 5 Aug 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
CSSNET: A LEARNING ALGORITHM FOR THE SEGMENTATION OF COMPRESSED HYPERSPECTRAL IMAGES

M. Biquard, A. Rouxel*, S. Lacroix, H. Carfantan†, A. Monmayrant, H. Camon

LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France

ABSTRACT

The paper presents a semantic segmentation method which is directly applicable to compressed hyperspectral images acquired with a dual-disperser CASSI instrument. It introduces an algorithm based on a shallow neural network that exploits the spectral filtering performed by the optical system and the compressed hyperspectral images measured by the detector. Encouraging results that exploit 50 to 100 less data than the whole hyperspectral datacube on PaviaU and IndianPines datasets are presented.

Index Terms—Compressed hyperspectral imaging, Dual-Disperser CASSI, Compressed images segmentation

1. INTRODUCTION

Classic hyperspectral imagers that produce a whole data cube pose a twofold challenge in environmental observation and monitoring from space, firstly because of the quantity of data to be transmitted to the ground, and secondly because of the resources required for post-processing analyses. Compressed hyperspectral acquisition systems, where each pixel of the acquired images combines several spectral components, can alleviate these issues, firstly by reducing the amount of data to be acquired and transmitted.

The interpretation of compressed measurements requires the use of algorithms that exploit various assumptions about the observed scene, such as regularity and redundancy, and an a priori knowledge of the spectral filtering performed by the optical system. The classical approach is to reconstruct the complete hyperspectral cube from the compressed data. State-of-the-art methods such as TwIST [1], GAP-TV [2], DeSCI [3] and \( \lambda \)-net [4] obtain impressive results that rely on theoretical work on “Coded-Aperture Snapshot Spectral Imager” systems (CASSI, [5, 6]). Nevertheless, in many cases and particularly in remote sensing, the objective of image acquisition is not to obtain the signal in its raw form, but rather to extract given semantic information from the scene. In such cases, the reconstruction is only an intermediate step in data processing, it can be costly by itself, it still requires post-processing of the reconstructed cube, whereas it can by-passed by directly processing the compressed data. In this paper, we present a semantic segmentation method which is directly applicable to compressed hyperspectral images acquired with a dual-disperser CASSI instrument. By refraining from reconstructing the raw signal, this work is in line with work done by Davenport [7, 8], Arguello [9, 10, 11] and Arce [12]. We propose an algorithm based on a shallow neural network that exploits the spectral filtering performed by the optical system and the compressed hyperspectral images measured by the detector.

2. DUAL-DISPERSER CASSI

Our proposal is based on the use of a DD-CASSI hyperspectral imager, originally introduced in [13], on the basis of which we have developed a prototype composed of two symmetrical spectro-imagers on both sides of a micro-mirror array (DMD) [14]. The first spectro-imager spatially separates and images the different spectral planes of the scene on the DMD. Each spectral plane is therefore filtered by a different piece of the DMD. The second spectro-imager recombines the filtered spectral planes and re-images them on the camera, cancelling any spectral dispersion (figure 1). The essential property of this system and necessary to our approach is the co-localization: whatever the spatial filtering introduced by the DMD, the spectral components present at a point \((x, y)\) of the camera come only from the spectrum present in the scene at the corresponding point. A consequence of co-localization is a direct access to the panchromatic image on the camera by completely opening the DMD.

In the more general case of any filtering pattern on the DMD, we measure at each point of the camera a weighted sum of the different components of the spectrum at this point (equation 1). The weights are directly related to the mask programmed on the DMD and the dispersion of spectral planes on it. It is thus possible from the mask and the dispersion of the system to build a filtering cube.

The discretized model of the DD-CASSI imager is expressed as follows:

\[
I_{r,c} = \sum_{k=0}^{n_A-1} h_{r,c,k}S_{r,c,k}
\]

with \( I_{r,c} \) the intensity measured by the pixel \((r, c)\) of the...
detector, \( r \in [0, n_r - 1] \) and \( c \in [0, n_c - 1] \) with \( n_r \) and \( n_c \) the numbers of rows and columns on the detector. \( n_{r,c,k} \) is the number of spectral components. \( h_{r,c,k} \) is the value of the filter cube associated with the \( k \)-th spectral component of the pixel \((r, c)\) and \( S_{r,c,k} \) is the spectro-spatial intensity of the observed scene, associated with the voxel \((r, c, k)\).

3. CSSNet Architecture

3.1. Related work

A number of works aim at extracting semantic information from the full hyperspectral cube via neural networks. Given the lack of labeled cubes, this extraction is done from small patches of the scene. Approaches come to classify them using 1D and 2D convolutions [15] [16] [17], or 3D convolutions [18] [19]. Others do semantic segmentation on these patches [20].

Recent works seek, as we do, to segment the scene directly from the compressed images. Approaches inspired by compressed sensing segment the scene based on sparsity assumptions [21]. Others rely on Bayesian probabilities to propose iterative segmentation [22]. Hao Zhang et al. use neural networks to co-optimize scene segmentation and mask generation [23].

3.2. Network inputs

The network works with patches of size \( p \). To form the input, we need 3 objects: the filter cube with spatial dimensions of the patch \( H \in [0, 1]^{p \times p \times n_{\lambda}} \), the associated compressed image \( I \in \mathbb{R}^{p \times p} \) and the panchromatic image of the scene \( P \in \mathbb{R}^{p \times p} \). The output is \( O \in \mathbb{R}^{p \times p \times q} \) with \( q \) the number of classes. We estimate \( S \in \{0, 1, ..., q-1\}^{p \times p} \) with \( \hat{S} = (\arg \max O_{i,j})_{0 \leq i,j \leq p-1} \).

We want to extract the spectral information of \( I \) by removing \( P \). But \( I \) is only a percentage of the light flux of \( P \) and does not have the same variations. We then define \( I_{\text{spec}} = \hat{I} - \hat{P} \) with \( \hat{I} \) and \( \hat{P} \) the normalized versions of \( I \) and \( P \).

The input of the network consists of \((\hat{H}, \hat{I}_{\text{spec}}, \hat{P})\), \( \hat{H} \) and \( \hat{I}_{\text{spec}} \) being the normalized versions of \( H \) and \( I_{\text{spec}} \).

3.3. Network architecture

The CSSNet (“Compressed Semantic Segmentation Network”) is divided into two parts. The first one processes the spectral information brought by the compressed image and the filtering cube. The second part mutualizes these data with the spatial information extracted from the panchromatic image.

The network is only made of convolution layers on feature maps of size \( p \times p \): a sufficient padding is applied to keep the size of these feature maps.

A detailed diagram of the architecture is given in Figure 2.

**Spectral processing.** This part includes two blocks of convolutions \( F_1 \) and \( F_2 \). The treatment performed is:

\[
H_2 = F_1(\hat{H}), \quad H_3 = F_2([H_2, \hat{I}_{\text{spec}}])
\]

where the operation \([., .]\) represents the concatenation along the channels. \( \hat{H} \) is of dimension \( p \times p \times n_{\lambda} \) : we consider for the input of \( F_1 \) that \( \hat{H} \) is of dimension \( p \times p \) with \( n_{\lambda} \) channels.

**Spatial/spectral processing.** The architecture used is inspired by the DSSNet [20] used for the segmentation of the complete hyperspectral cube, to which two convolution layers have been added, and the number of channels modified. This part is decomposed into three convolution blocks \( F_3, F_{3,\text{sym}}, \) and \( F_4 \).

\[
H_4 = F_3(H_3), \quad H_5 = F_{3,\text{sym}}([H_4, \hat{P}]), \quad O = F_5(H_5)
\]

**Optimisation.** The loss used is the weighted cross entropy loss to evaluate the difference between \( \hat{S} \) and \( S \):

\[
\mathcal{L}_{CE} = -\frac{1}{N} \sum_{s=1}^{N} \sum_{k=0}^{q-1} w_k \sum_{0 \leq i,j \leq p} t_{i,j,k}^s \log(p_{i,j,k}^s)
\]

with \( N \) the number of patches, \( p_{i,j,k}^s \) the softmax probability that pixel \((i, j)\) of patch \( s \) belongs to class \( k \), \( t_{i,j,k}^s \) the class label \( k \) for pixel \((i, j)\), and \( w_k \) the inverse of the median frequency, computed over the whole training dataset, of class \( k \).
Fig. 2. Architecture of the CSSNet network. Each blue, yellow, green or red rectangle represents a convolution layer. The number inside each convolution layer represents the number of channels. The inputs of the network are: the patch “panchromatic image” normalized $\bar{P}$, the patch “compressed image” normalized $\bar{I}_{\text{spec}}$ and the patch “associated filtering cube” normalized $\bar{H}$. The output of the network is $\hat{S}$ which associates to each pixel of the patch a vector of probabilities.

Fig. 3. Used datasets

### 4. RESULTS

#### 4.1. Methodology

**Datasets used.** The network is tested on two datasets: Indian Pines [24] and Pavia University [25]. To build the database, these two hyperspectral cubes are run through a DD-CASSI system simulator. The characteristics of the two datasets are presented in Figure 3.

**Training and inference separation.** Training and inference are performed on compressed images from the same hyperspectral cube. The experiments are done using a well defined separation in which no pixel of the hyperspectral cube involved in the inference is seen during the training phase. An area of 10% is reserved for inference, 80% for training (the rest of the patches are located at the border).

**Experimental settings.** The patches are of size $7 \times 7$. For each dataset, 40 compressed images and one panchromatic image are acquired. 80% of the compressed images are used for training and 20% for inference. For inference, the image is scanned with a sliding window of stride 3. The network is trained on 60 epochs with "early stopping". The learning rate is 0.01, the batch size is 100, the weight decay is $5 \times 10^{-4}$. The optimizer used is SGD (Stochastic Gradient Descent) with a momentum of 0.9.

**Metrics.** Two metrics are used: the accuracy (OA) and the average f-score (average of the f-scores per class).

**Comparison to existing work.** As the implementation details leading to the results presented in [23] are not explicit, we only compare ourselves to segmentation networks on full hyperspectral cubes: the 3D CNN of [15] implemented by [26] and DSSNet (our own implementation).

#### 4.2. Segmentation results

![Fig. 4. Comparison of segmentation algorithm results on PaviaU and Indian Pines. The two shaded lines correspond to the reference algorithms to which we compare ourselves. Unlike our networks, these algorithms take the entire hyperspectral cube as input. The next two rows of the table contain the results of two different versions of our segmentation network. CSSNet (mix) corresponds to a version of CSSNet where the panchromatic image $\bar{P}$ is introduced together with the compressed image $\bar{I}_{\text{spec}}.$](image)
Each network has been trained 5 times, and the average of the scores is displayed in Figure 4. The underperformance of 3D-CNN on Pavia University, representing the state of the art, is probably due to a problem - whose source we did not find - of hyperparameter setting.

For each CSSNet training, the metrics are computed for several DMD masks and the average is taken. Note that with the best performing masks, the accuracy on Pavia University averages 86.5.

Compared to the full cube segmentation, there is a loss of performance which is natural because we process 50 and 100 less data volume on Pavia University and Indian Pines. But the results are still acceptable, the details of the errors are shown in Figure 5.

![Fig. 5. Segmentation results for DSSNet and CSSNet. Each color represents a class, unlabeled areas are in black. Right: the test areas exploited for the numerical evaluations presented Table 4.](image)

**On spatial-spectral separation.** \( \hat{P} \) is introduced only at the end of the network. A consequent part of this one is dedicated only to the extraction of spectral information. We observe, by introducing \( \hat{P} \) at the same time as \( \hat{I}_{\text{spec}} \) (CSSNet (mix) in Figure 1), a less good generalization of the network to inference. The assumption made is that extracting spectral information from \( \hat{H} \) and \( \hat{I}_{\text{spec}} \) is a more complex task than extracting contours from \( \hat{P} \). By incorporating \( \hat{P} \) earlier, the network gives it too much importance and “over-learns” the spatial part.

**5. CONCLUSION**

In this paper, we have presented an algorithm for semantic segmentation from compressed hyperspectral data. It is based on a neural network architecture taking into account the specificities of the measurement system. Simulation results indicate lower performances than the state of the art on full hyperspectral cube but the number of acquisitions required to obtain the measurements is drastically reduced by nearly two orders of magnitude. The next steps are to perform iterative segmentation, either with Bayesian fusion or using a RNN type architecture, and perform tests on data sets acquired with our prototype [14].

**6. REFERENCES**


[25] Paolo Gamba, “Pavia University hyperspectral dataset from Grupo de Inteligencia Computacional,”
