Multivariate Image Fusion: A Pipeline For Hyperspectral Data Enhancement

João Fortuna[a,b,1,*], Harald Martens[a,c], Tor Arne Johansen[a,b]

[a]*Norwegian University of Science and Technology (NTNU), Department of Engineering Cybernetics, Trondheim, Norway*
[b]*Centre for Autonomous Marine Operations and Systems (NTNU AMOS), Trondheim, Norway*
[c]*Idletechs AS, Trondheim, Norway*

Abstract

Hyperspectral cameras provide high spectral resolution data, but their usual low spatial resolution when compared to color (RGB) instruments is still a limitation for more detailed studies. This article presents a simple yet powerful method for fusing co-registered high spatial and low spectral resolution image data – e.g. RGB – with low spatial and high spectral resolution data – Hyperspectral. The proposed method exploits the overlap in observed phenomena by the two cameras to create a model through least square projections. This yields two images: 1) A high-resolution image spatially correlated with the input RGB image but with more spectral information than just the 3 RGB bands. 2) A low-resolution image showing the spectral information what is spatially uncorrelated with the RGB image. We show results for semi-artificial benchmark datasets and a real-world application. Performance metrics indicate the method is well suited for data enhancement.

Keywords: Hyperspectral, Data Fusion, Pansharpening, Super Resolution

1. Introduction

Information on distribution and abundance of natural resources is important for science, education, policy making and management alike [1]. Hyperspectral (HS) instruments provide a richness of data that enables classification and detection of such resources

*Corresponding author
Email address: joao.fortuna@ntnu.no (João Fortuna)

[1]This work was supported by the Norwegian Research Council (grant no. 223254) through the Centre of Autonomous Marine Operations and Systems (NTNU AMOS) at the Norwegian University of Science and Technology, the MASSIVE project (grant no. 270959), as well as the Norwegian Space Center.

through passive and non destructive measurements. Furthermore, they can scan large extents of ground in a short time period, making them well suited tools for air- and space-borne remote sensing. Low-cost hyperspectral systems recently developed [2] make this technology more accessible to research groups all over the world. While such low-cost systems are not expected to produce the same high quality data as more expensive equipment, they may be well suited for certain applications.

Using multiple sensors with different capabilities, often creates a clearer picture of the environment when compared to a single sensor scenario. However, while our brains are good at fusing information from different sources, some work is required if the process is to be automated. Multivariate calibration, a term coined in the field of chemometrics, refers to the development of models to explain certain properties of interest by combining different variables from multi-channel sensor measurements.

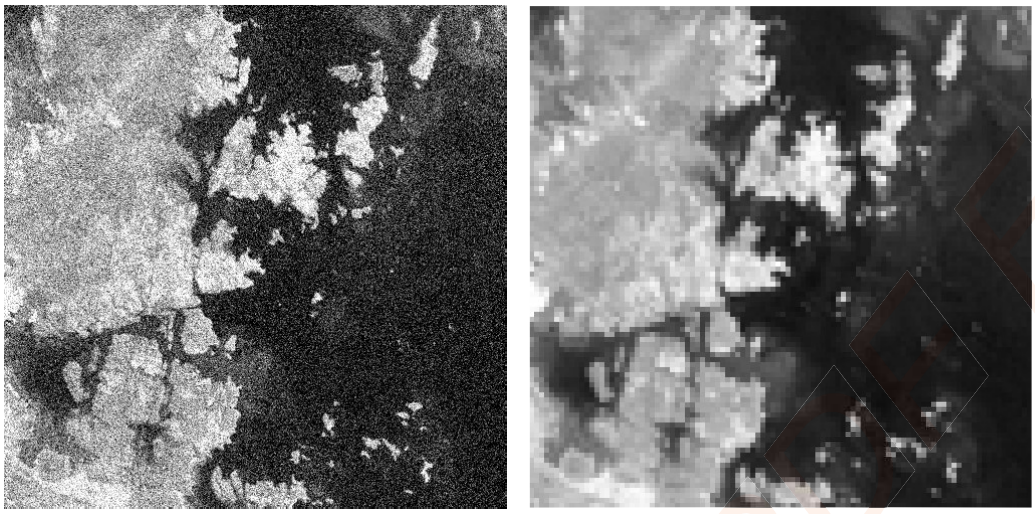
Fusing images from sensors with different spectral and spatial properties to generate a single, improved data product is a known and studied problem [3, 4]. Hyperspectral image super-resolution fusion methods can be grouped into 4 categories: Bayesian based approaches [5–11]; Tensor based approaches [12–17]; Matrix factorization based approaches [18–33]; and Deep Learning based approaches [34–41].

The mentioned methods assume that the images to be fused are co-located (registered). However, more recent methods drop that assumption and achieve simultaneous registration and super-resolution [42, 43].

1.1. There is no such thing as a free lunch, or photons

Even though **number of pixels** is a different concept from **spatial resolution**, they are tightly coupled. Say we have two monochrome focal plane array (FPA) cameras, *A* and *B*, on a satellite and both image the same area on the ground, see Figure 1. If *A* has 5 times the amount of pixels in both axis, then the spatial resolution – the ability to differentiate between two close objects, or in this case the size of one pixel on the ground – will be 5 times better than the one of camera *B*. Here we assume all other properties of the cameras, such as optics, and image capture to be the same and that they do not limit the resolution, i.e. the sensor is the bottleneck.

Higher resolution has the obvious benefit of allowing finer details to be seen, however,



(a) High resolution (500×500 px) (b) Low resolution (100×100 px)

Figure 1: Example satellite images with simulated different resolutions. Sensors have same area but different number of pixels (different pixel size and density). The higher number of pixels in (a) means less light (fewer photons) per pixel, hence more noise. Photo by NASA on Unsplash [44].

increasing the number of pixels is not always possible, particularly as we increase the number of spectral bands. To understand why the number of bands affects the spatial resolution, we need to think of light reaching a camera as a stream of a finite number of photons. These particles need to travel through the optical elements of the camera, then they are distributed by all the pixels in the sensor, where they are transformed into an electrical current that is finally converted into digital data. Higher pixel density means less photons per pixel, as we divide the same finite amount of photons by a higher number of pixels, and fewer photons per pixel leads to a weaker signal (low SNR). When the signal is fainter, the noise contribution becomes apparent if we increase the sensor gain, compare (a) and (b) in Figure 1. If we now try to sort the photons into many spectral bands, they become even scarcer and we need to compromise on the number of spatial pixels in order to still have a usable signal.

Throughout this article we will mention high and low resolution data, in this scope we mean both the number of pixels and ability to resolve a smaller object in the image. Higher resolution data will have more pixels and conversely, lower resolution, fewer pixels.

Panchromatic images contain information from a broad spectrum in a single band, hence

can more easily have a high spatial resolution. Pansharpening methods were initially developed in the mid 1980s for air- and space-borne multispectral imagers with low resolution that could be improved with high resolution panchromatic images. With the ever growing availability of hyperspectral instruments, some of those methods were adapted to hyperspectral data and others developed anew. Several methods to achieve such sharpening are described in the literature [3].

Thanks to advances in sensor technology, we now have spatially high resolution color cameras (Red-Green-Blue – RGB) and even some multispectral cameras (with few, but more than 3, bands), which we can use instead of panchromatic when enhancing hyperspectral data. The advantages of using colour cameras are clear: Even with only 3 bands we have multivariate – as opposed to univariate – spectral data in high spatial resolution. Such multivariate data gives much better selectivity, as it adds color information to the simple measurement of light intensity provided by panchromatic sensors. This is important in the present setting, where a spatially high-resolution RGB camera is combined with a spatially low-resolution hyperspectral camera: The higher number of bands with high spatial resolution, the more we can improve the spatial resolution of hyperspectral data, assuming that both cameras have recorded the same spatial scene and therefore can be correlated.

Some of the most commonly used pansharpening methods are useful when the intended goal is to produce a high resolution RGB image from multi-/hyperspectral data, having only a high resolution monochromatic image. However, those enhancements are not visible when the goal is, for example, to find high resolution estimated abundance maps of geological or biological resources of interest.

1.2. Motivation

For the past years, our research group has been working on a lightweight hyperspectral imaging system for unmanned aerial systems (UAS) [2][45]. Because of weight, cost and complexity limitations, and operating conditions, the spatial resolution has been the most limiting factor when it comes to generating high quality data products. RGB cameras are very often already a part of the payload carried by such unmanned systems, and if not, they are a simple addition.

Taking that into consideration, trying to improve HS spatial resolution with RGB data was set as the goal for the present research.

1.3. Paper contribution

In this paper we describe a generic framework for multivariate image fusion, building on the ideas of pansharpening while trying to also enhance the output for further processing, instead of just visual representation. Hence, the purpose of the present methodology is to use the high spatial resolution of the RGB measurements to yield an equally high resolution representation of the low resolution HS measurements. Other studies [46], [47] have pursued somewhat related approaches, nevertheless,the method proposed here is to the authors' knowledge, a new development. It is fast, when compared to other methods in the literature [3], and requires very little knowledge of calibration parameters or relationship between the two datasets to be fused, only assuming that they have been spatially registered beforehand. In summary, the present method combines the input, consisting of a low-resolution multi-channel HSI image and a high-resolution 3-channel RGB image of the same scene, into two output images: 1) A high-resolution multi-channel image showing what is spatially correlated to the RGB image, and 2) A low-resolution multi-channel image showing what is spatially uncorrelated to the RGB image. The following section describes each step of the method. We then show some results of applying the method to both artificially degraded real data, and real low resolution data. To conclude, a discussion on possible improvements and future work.

2. Method - Multivariate Image Fusion (MVIF)

In this section we present a method for fusing RGB and HS data in order to get a data product that takes advantage of the relative strengths of both, Figure 2. Such fusion is possible because HS data is in most real world applications typically very rank deficient — the number of HS wavelength bands is much higher than the number of statistically independent spectral variation types in the image. That means it is possible to learn all relevant patterns of variation in HS and replicate them through a low rank, but high spatial resolution, approximation based on RGB.

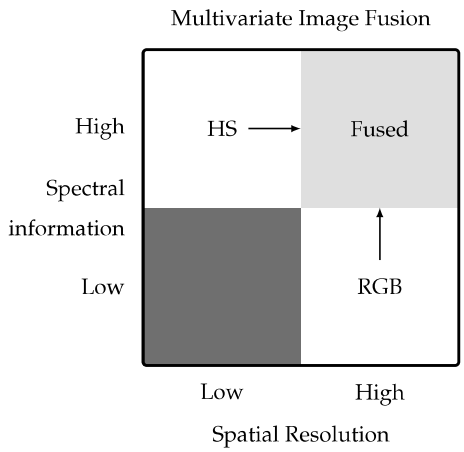


Figure 2: Multivariate Image Fusion exploits the relative strengths of RGB and Hyperspectral data.

2.1. Data Model

Before continuing, it is useful to write down the data model we will be working with. We take a similar approach as [48], based on the hypothetical model:

$$\boldsymbol{Y} = \boldsymbol{C}\boldsymbol{S}^{\mathsf{T}} + \boldsymbol{D}\boldsymbol{Z}^{\mathsf{T}} + \boldsymbol{F} \tag{1}$$

In this model we assume that data cubes are unfolded, so all elements are 2D matrices.
$\boldsymbol{Y}$ is a high spectral resolution (hyperspectral, here called "full-spectral") image dataset, with high or low spatial resolution, depending on the context.
$\boldsymbol{C}$ is the RGB data. The product $\boldsymbol{C}\boldsymbol{S}^{\mathsf{T}}$ contains the known spatial pattern from the RGB camera ($\boldsymbol{C}$), based on initially unknown, but estimated full-spectral ($\boldsymbol{S}^{\mathsf{T}}$) information.
In analogy to the previous element, $\boldsymbol{D}\boldsymbol{Z}^{\mathsf{T}}$ contains spatial ($\boldsymbol{D}$) and spectral ($\boldsymbol{Z}^{\mathsf{T}}$) information, but now from phenomena that are not seen by the RGB camera. Initially unknown, ($\boldsymbol{D}$) and spectral ($\boldsymbol{Z}^{\mathsf{T}}$) have to be estimated.
$\boldsymbol{F}$ will, ideally, contain only noise.
When dealing with remote sensing spectral data, it is helpful to think of the total signal in each pixel of $\boldsymbol{Y}$ as a sum of contributions from all the phenomena that were observed in that single pixel. These contributions have 2 properties: concentration/abundance $\boldsymbol{C}$ and $\boldsymbol{D}$ and spectral signature $\boldsymbol{S}$ or $\boldsymbol{Z}$. When we consider all the pixels in $\boldsymbol{Y}$, the concentrations become spatial distribution maps of each of those phenomena. Adding all the spectral signatures – $\boldsymbol{S}$ and $\boldsymbol{Z}$ – weighted according to their respective concentration per pixel – $\boldsymbol{C}$

and $\boldsymbol{D}$ – yields the signal in $\boldsymbol{Y}$, aside from noise – $\boldsymbol{F}$:

$$\boldsymbol{Y} = [\boldsymbol{C}, \boldsymbol{D}]\,[\boldsymbol{S}, \boldsymbol{Z}]^{\mathsf{T}} + \boldsymbol{F} \tag{2}$$

Even though the number of bands (spectral resolution) can be of several tens or even hundreds, hyperspectral data is typically rank deficient, which means we can obtain a much lower dimension representation with less noise, while still keeping the relevant information. On the other hand, RGB data usually has a full rank of 3 in the spectral domain. When the spectral range of both instruments is overlapping, they observe the same phenomena and we can obtain a low rank representation of HS using RGB data.

If we use RGB data as is – with 3 bands – we are limited to a 3 dimension low rank representation, and while this may be enough for some datasets, it will prove inadequate for more complex scenes. This is a limitation of using a linear projection method as we have done here, other methods may not face this problem. Fortunately, it is possible to artificially expand the number of high resolution bands by appending the result of non-linear operations on the original RGB data, thus increasing the rank of the high resolution data. Such operations are for example: interaction terms (product of different bands) or square terms:

$$\boldsymbol{C}_{RGB} = [\boldsymbol{C}_R, \boldsymbol{C}_G, \boldsymbol{C}_B] \tag{3}$$

$$\boldsymbol{C}_{int} = [\boldsymbol{C}_R \circ \boldsymbol{C}_G, \boldsymbol{C}_R \circ \boldsymbol{C}_B, \boldsymbol{C}_G \circ \boldsymbol{C}_B] \tag{4}$$

$$\boldsymbol{C}_{sqr} = [\boldsymbol{C}_R \circ \boldsymbol{C}_R, \boldsymbol{C}_G \circ \boldsymbol{C}_G, \boldsymbol{C}_B \circ \boldsymbol{C}_B] \tag{5}$$

$$\boldsymbol{C}_{ext} = [\boldsymbol{C}_{RGB}, \boldsymbol{C}_{int}, \boldsymbol{C}_{sqr}] \tag{6}$$

If we consider $\boldsymbol{C}_{RGB}$ – Eq. 3 – to be the original RGB-only data, unfolded, where each column represents a color channel, then we can define a matrix of first-degree interactions, $\boldsymbol{C}_{int}$, and another of square terms, $\boldsymbol{C}_{sqr}$, respectively by element-wise multiplying each band by another, Eq. 4, or by itself, Eq. 5. The operator $\circ$ represents the element-wise multiplication, also known as Hadamard product. When composing the $\boldsymbol{C}$ matrix to input to the algorithm, we could use $\boldsymbol{C}_{ext}$ if we wanted to include interaction and square terms. Fundamentally, interaction and square terms do not add new information, however by

providing these non-linear terms to the linear algorithm, it allows it to find non-linear spectral variations, which are expected to exist. This is analogous to how different wavelength channels have different non-linear relations to the chemical sample composition in NIR multichannel reflectance measurements. These unknown but different non-linear relations may be regarded as a special type of unknown interference. Using the pragmatic but incorrect $\log(1/R)$ transform allows linear multivariate calibration modelling, e.g. by PLSR to utilize the additional subspace dimensions, spanned by the channels' unknown differences in non-linearity, to pick up and correct for these unknown interference, as described in [49].

For our present RGB data we do not know the detailed camera properties. In addition, the light signal is affected by the atmospheric absorbance and light scattering effects in the water phase that the photons have to go through, on their way from the light source, the Sun, via the bottom object and back to the camera. We do not know the ideal mathematical transform from chemical and physical properties of the objects on the bottom, to the RGB signal of the camera, but the transform $\log(1/R)$ is probably too simplistic. However, by adding new "wavelength channels" by non-linear combinations, e.g. interactions and square terms, of the original RGB channels, the linear multivariate calibration has a better chance of finding a subspace that spans both the chemical and physical signal variations and their non-linearities.

Later, when we discuss the performance of our proposed method, we show how different combinations of non-linear terms affected it.

2.2. Notation

Throughout the paper we will use the following notation:

- Unfolded HS data cube with k_H bands and low spatial resolution $n_L^h \times n_L^w$ — $\boldsymbol{Y}_L \in \mathbb{R}^{n_L \times k_H}$
- Unfolded RGB data (and appended artificially generated terms) with k_L bands and high spatial resolution $n_H^h \times n_H^w$ — $\boldsymbol{C}_H \in \mathbb{R}^{n_H \times k_L}$
- Enhanced HS data with high spectral and spatial resolutions — $\widehat{\boldsymbol{Y}}_H \in \mathbb{R}^{n_H \times k_H}$

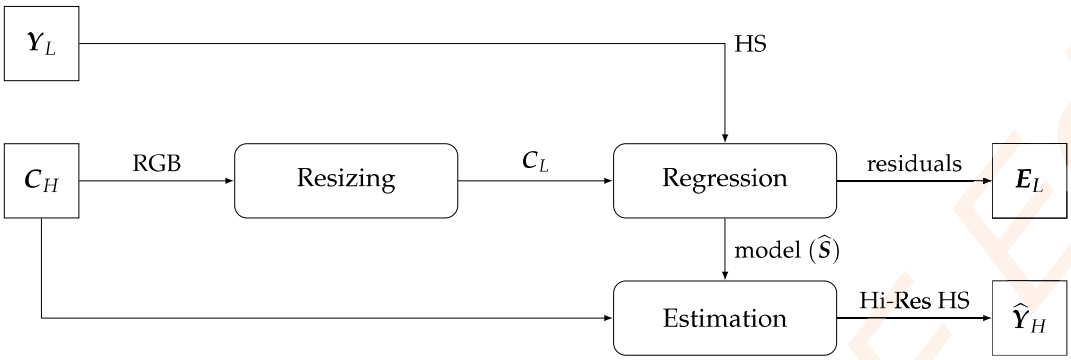


Figure 3: Pipeline overview diagram.

Where $n_L = n_L^h \times n_L^w$ and $n_H = n_H^h \times n_H^w$ are respectively the total number of spatial pixels – height (h) and width (w) – in low and high resolution data.

2.3. *Method overview*

The algorithm can be summarized in the following steps, also visible in Figure 3:

1. **Resizing** (shrinking) high-resolution RGB data to low-resolution HS size, with image registration, to ensure that the pixels in both images represent the same ground positions.
2. Noise weighted modelling to estimate HS from low-resolution RGB by regression over the low resolution pixels, through **Regression**. Lack-of-fit HS residuals are kept for further analysis of spectral patterns not seen in RGB.
3. **Estimation** of HS using high-resolution RGB.

2.4. *Resizing*

As mentioned before, $\boldsymbol{Y}_L$ and $\boldsymbol{C}_H$ have different image dimensions, so we start by shrinking $\boldsymbol{C}_H$ for it to coincide $\boldsymbol{C}_L$. Resizing should take into account the properties of the HS instrument that resulted in such low resolution – i.e. if sampling frequency is low but exposure time is also low, resulting in a subsampled target, then we should resample the high resolution RGB data. If on the other hand, the exposure time is long and the target (ground) is fully sampled/observed, but the information is mixed/convolved due to motion blur in each pixel, then we should apply a similar convolution to mix the RGB pixels. This way we ensure that the ground contributions are similarly represented between the HS and RGB data.

2.5. Noise-balancing wavelength weights

Hyperspectral instruments have varying levels of noise for each band. The proposed methodology involves least-squares based estimation of parameters in a reduced rank regression model. For such methods it is important to balance the noise level of the different wavelength channels. For the most common pushbroom slit-grating design the noise is generally worse as we move away from the center of the sensor and the operation range of the (electro-)optical components. In VIS-NIR instruments, usually with CMOS sensors, performance degrades quickly for bands below 400nm and above 900nm, but even inside that range, the noise level varies from channel to channel. Knowing how the noise varies improves the modeling performance, by down-weighing noisy bands we reduce the risk of over fitting noise. We estimated the noise according to the method described in Appendix A.1.

2.6. Regression and Estimation

Once pre-processing is done, we can proceed to the core of our method, the regression step. Here we estimate $\boldsymbol{S}$, knowing $\boldsymbol{Y}$ and $\boldsymbol{C}$. Generically speaking, we establish a projection model:

$$\boldsymbol{Y} = \boldsymbol{C}\boldsymbol{S}^{\mathsf{T}} + \boldsymbol{E} \tag{7}$$

Then we can apply the previous equation to our data:

$$\boldsymbol{Y}_L = \boldsymbol{C}_L\boldsymbol{S}^{\mathsf{T}} + \boldsymbol{E}_L \tag{8}$$

$$\widehat{\boldsymbol{S}}^{\mathsf{T}} = (\boldsymbol{C}_L^{\mathsf{T}}\boldsymbol{C}_L)^{-1}\boldsymbol{C}_L^{\mathsf{T}}\boldsymbol{Y}_L \tag{9}$$

$$\widehat{\boldsymbol{Y}}_L = \boldsymbol{C}_L\widehat{\boldsymbol{S}}^{\mathsf{T}} \tag{10}$$

$$\boldsymbol{E}_L = \boldsymbol{Y}_L - \widehat{\boldsymbol{Y}}_L \tag{11}$$

The matrix $\widehat{\boldsymbol{S}}$ contains the estimate of a dictionary that translates the variations in $\boldsymbol{C}_L$ into variations in $\boldsymbol{Y}_L$. Furthermore, we can now use $\boldsymbol{C}_H$ with that same dictionary and compute:

$$\widehat{\boldsymbol{Y}}_H = \boldsymbol{C}_H\widehat{\boldsymbol{S}}^{\mathsf{T}} \tag{12}$$

which gives a high spatial resolution estimation of $\boldsymbol{Y}$.

2.7. Low resolution residuals analysis

When estimating $\widehat{\boldsymbol{S}}$ and $\widehat{\boldsymbol{Y}}_L$, we are left with unmodelled low-resolution residuals – $\boldsymbol{E}_L$. These can be analysed to give us some insight into what could not be enhanced to higher resolution – systematic information not captured by the model – and estimate how much of it was random independent noise. In order to do that, we need to further decompose $\boldsymbol{E}_L$, through some bilinear matrix decomposition techniques, according to the model:

$$\boldsymbol{E}_L = \boldsymbol{D}_L \boldsymbol{Z}^\intercal + \boldsymbol{F}_L \tag{13}$$

$$\boldsymbol{D}_L \in \mathbb{R}^{n_L \times A} \tag{14}$$

$$\boldsymbol{Z} \in \mathbb{R}^{k_H \times A} \tag{15}$$

where A is the number of factors, or components extracted.

Referring back to our overall model in Eq. 1, then $\boldsymbol{D}_L$ and $\boldsymbol{Z}$ contain low-resolution spatial and respective spectral information of phenomena that are not measured by the RGB camera. $\boldsymbol{F}$ contains unmodeled noise, in low-resolution.

2.7.1. Matrix decomposition

Matrix decomposition or factorization, also called unmixing in the context of spectral data, is a family of methods that split a matrix in a product of other matrices. For hyperspectral data, those resulting matrices usually correspond to some type of spectral signatures and respective spatial distribution and/or concentration.

A simple yet useful factorization method is Singular Value Decomposition (SVD) [50][49]. However, a property of the resulting spectral features is that they are orthonormal, hence not directly representative of bio/geo/chemical spectral signatures.

On the other hand, with Non-Negative Matrix Factorization (NNMF) [51], Multivariate Curve Resolution (MCR) [52] or Independent Component Analysis (ICA) [53], the spectral components (loadings) are often related to actual spectral signatures of phenomena seen in the captured scene. This comes at the expense of more complex computation.

Hyperspectral data is notoriously rank deficient, meaning that factorization methods will model a limited number of meaningful components and many noise components.

For the implementation of the MVIF pipeline here described we first use SVD to estimate

the number of non-noise components (A) in the residuals, then we can use one of the more complex methods knowing how many components to expect.

There is not a consensus among specialists regarding which is the best method to select the appropriate number of relevant components when using SVD [54], furthermore many of them require visual inspection of plots. We propose a solution based on a voting system: three methods evaluate different metrics and vote on whether a component is relevant or not. If a component gets the all votes, it is deemed relevant. Find more details about this method in Appendix Appendix A.3.

Once we have all the votes from the 3 classifiers, we decide how many factors to keep – A. Then it is simply a matter of running the unmixing method of our choice, to obtain the factorization as in Equation 13.

$$\boldsymbol{E}_L \xrightarrow[\text{decomp}]{\text{matrix}} \widehat{\boldsymbol{D}}_{L,A}, \widehat{\boldsymbol{Z}}_A \tag{16}$$

$$\widehat{\boldsymbol{E}}_{L,A} = \widehat{\boldsymbol{D}}_{L,A}\widehat{\boldsymbol{Z}}_A^{\mathsf{T}} \tag{17}$$

$$\boldsymbol{F}_{L,A} = \boldsymbol{E}_L - \widehat{\boldsymbol{E}}_{L,A} \tag{18}$$

3. Results

In this section we show and analyse the results of applying the presented method to two distinct datasets. In addition, we also compare metrics for a third benchmark dataset. First, we generate low resolution hyperspectral data by degrading a high resolution hyperspectral data cube. RGB data is also extracted from the high resolution HS data. As we have a high resolution reference, we can quantify the performance of the algorithm using the performance metrics described in [3].

Second, we use a sample of data from a UAV field campaigns, for which this method was conceived for. RGB comes from a separate camera. Since there is no high resolution reference, performance can only be evaluated visually.

Finally, the benchmark dataset is just briefly analysed in order to compare the performance of MVIF to that of another method from literature on the same dataset.

(a) RGB representation of $\boldsymbol{Y}_H$, which in this case is also $\boldsymbol{C}_H$.

(b) RGB representation of $\boldsymbol{Y}_L$.

Figure 4: RGB rendering of HICO scene used during tests. The procedure described in Appendix A.2 was used to create both the high (500×500 px) and low resolution (100×100 px) images. Some color adjustment was applied to make the images more aesthetically pleasing.

3.1. Data

3.1.1. Control Dataset

We used data from the HICO instrument, available at [55]. HICO (Hyperspectral Imager for the Coastal Ocean) was a hyperspectral imager that was installed on the International Space Station (ISS) and captured data from 2009 to 2014. In terms of specifications, it has 87 bands (400-900 nm), cross-track resolution of 500 pixels and ground sample distance (GSD) of 90 m.

Both the RGB representation of the HS data and the reference RGB image were, for this dataset, extracted from the HS data – the usual procedure for benchmarking these types of algorithms. See Appendix A.2 for more details.

We used a sample of HICO data at full resolution as the high-resolution HS reference – dataset ID H2011145084342. From this dataset we extracted a 500×500 region of interest, see Figure 4. The low resolution HS data had 100×100 spatial pixels and 87 bands.

3.1.2. Test/Field Dataset

We also include the results of applying our method to data obtained in a field trial with drone mounted HS and RGB cameras, see Figure 5. This flight was conducted in Hopavågen, Norway, in March 2018. More details on the data capture and the experiment are available in [45].

In this dataset, the loss of spatial resolution was due to sub-optimal flight conditions, and instrument design limitations. The across-track resolution (here seen as the horizontal axis) is equivalent in both RGB and HS, but the along-track direction (vertical axis) is much lower for the HS camera. In fact, for each HS pixel, there are 30 RGB pixels.

Details about the spatial registration of the two images are outside of the scope of this paper. In broad strokes, a first coarse registration was possible due to timestamp synchronization between the two cameras, then fine-tuned through an image registration method – available in MATLAB as `imregister()`.

3.2. Performance metrics and benchmark dataset

As described in [3], there is a number of metrics commonly used to evaluate the performance of enhancement methods. Those metrics are: Cross Correlation (CC), measuring the spatial enhancement, with 1 as optimal value; Spectral Angle Mapper (SAM), as the name suggests indicates spectral fidelity, 0 is ideal; Root Mean Square Error (RMSE) and Erreur Relative Globale Adimensionnelle de Synthèse (ERGAS), both global quality indices, with 0 as ideal value. In Table 1 we compare how the values change for different choices of additional terms in $\boldsymbol{C}$ – both $\boldsymbol{C}_L$ and $\boldsymbol{C}_H$.

Table 2 shows the performance of MVIF in a benchmark dataset – Moffet field, also studied in [3] – compared to the best performing method mentioned in that publication. According to its authors, that performance is achieved in a machine with an Intel Core i5 3230M 2.6GHz with 8 GB RAM. Our method was running on an Intel Core i7 4510U 2.0GHz with 8 GB RAM, so an equivalent performance is expected. Even though the results seem better – slightly better values for the quality indices, and an extreme time reduction – we do have to say that our method uses an RGB reference, while the methods discussed in that publication are using a univariate – panchromatic – high resolution reference.

(a) High resolution RGB – $\boldsymbol{C}_H$.

(b) RGB representation of low resolution HSI – $\boldsymbol{Y}_L$ – stretched to same ratio of $\boldsymbol{C}_H$.

Figure 5: RGB image from ZenMuse camera (1500 × 560 px) and RGB rendering of HS data (50 × 560 px) from Hopavågen tests.

Table 1: Performance indices of MVIF with HICO dataset. Cross Correlation (CC), 1 is ideal. Root Mean Square Error (RMSE), Erreur Relative Globale Adimensionnelle de Synthèse (ERGAS) and Spectral Angle Mapper (SAM), 0 is ideal.

Terms in $\boldsymbol{C}$	CC	RMSE	ERGAS	SAM (deg)	Time (s)
RGB	0.823	0.081	10.339	6.888	1.751
RGB, Square	0.964	0.050	6.337	5.009	1.767
RGB, Square root	0.971	0.045	4.878	5.852	1.854
RGB, Interaction	0.970	0.043	5.485	4.686	1.849
RGB, Square, Square root	0.978	0.040	4.413	5.109	1.809
RGB, Interaction, Square, Square root	0.981	0.036	4.679	3.868	1.895

3.3. Plots

Results shown here were obtained taking into consideration the values in Table 1, meaning we opted for adding Interaction, Square and Square Root terms to $\boldsymbol{C}$ before sub-

Table 2: Performance comparison in Moffett field dataset. Cross Correlation (CC), 1 is ideal. Root Mean Square Error (RMSE), Erreur Relative Globale Adimensionnelle de Synthèse (ERGAS) and Spectral Angle Mapper (SAM), 0 is ideal. Values for Bayesian Sparse method extracted from [3]. To note that Bayesian Sparse enhanced HS data using only panchromatic high resolution data, while we used RGB.

Method	CC	RMSE	ERGAS	SAM (deg)	Time (s)
Bayesian Sparse	0.982	200.158	3.426	6.625	133.61
MVIF	0.985	164.861	3.424	5.427	0.95

mitting it to the Regression step, as it gave the overall best performance. In addition, we used Non-Negative Matrix Factorization (NNMF) to unmix the hyperspectral data cubes – $\mathbf{Y}_L$ and $\widehat{\mathbf{Y}}_H$ – and make it possible to represent in low dimension. Again, we estimated the number of factors using the method in Appendix A.3. The plots with the spectral signatures for each component are matched in color with the most similar between low – Figures 6 and 8 – and high – Figures 7 and 9 – resolution for each dataset.

Notice that each of the low resolution abundance maps are enhanced to high resolution, leaving no trace of low spatial resolution artifacts.

In Figures 10 and 11 we show a factorization of the residuals that could not be enhanced. Here we opted for using Independent Component Analysis (ICA), implemented as FastICA [56], instead of NNMF since the residuals are not non-negative. ICA gives a more interpretable factorization than SVD/PCA, while dealing well with possible, or in this case likely, negative concentrations. Another reason for using this method is that it is fast, and can give us a clue regarding whether there is relevant data that we overlooked, or not. If there is some indication that we should further analyse the residuals, other methods such as MCR can also be applied. We stress the importance of doing such complementary analysis on the residuals. This creates awareness regarding the limitations of the method, and even if low resolution, these are still relevant data.

4. Discussion

Correlation in noise for artificial datasets. When RGB and low resolution HS data are artificially generated for benchmarking, they both originate from the same high resolution HS reference. Naturally, there is a concern that noise in both is correlated. We have tried to

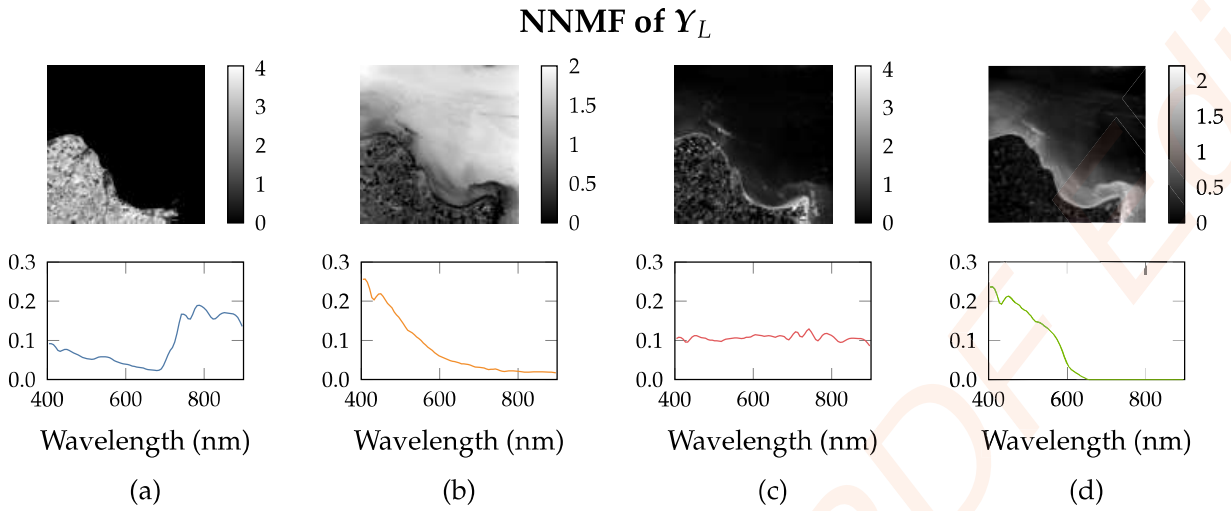


Figure 6: Non-negative Matrix Factorization (NNMF) of low resolution Y_L. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map. The components show here can be interpreted as land based vegetation (a), some combination of CDOM (color dissolved organic matter) and phytoplankton (b and d), and an albedo-like property of pixel (c) - almost flat spectrum.

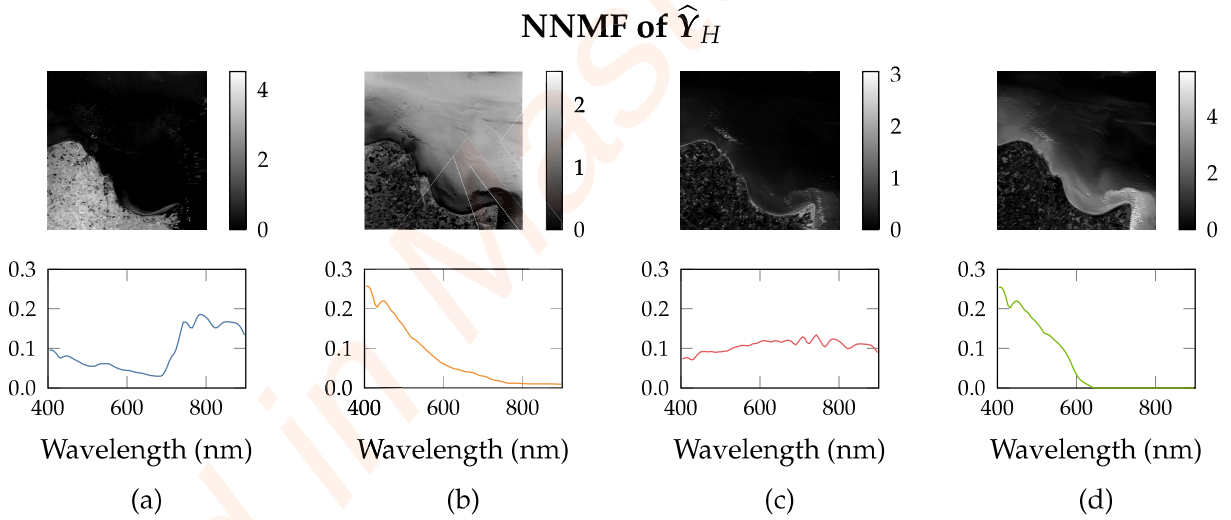


Figure 7: Non-negative Matrix Factorization (NNMF) of high resolution $\widehat{Y}_H$. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map. The components show here can be interpreted as land based vegetation (a), some combination of CDOM (color dissolved organic matter) and phytoplankton (b and d), and an albedo-like property of pixel (c) - almost flat spectrum.

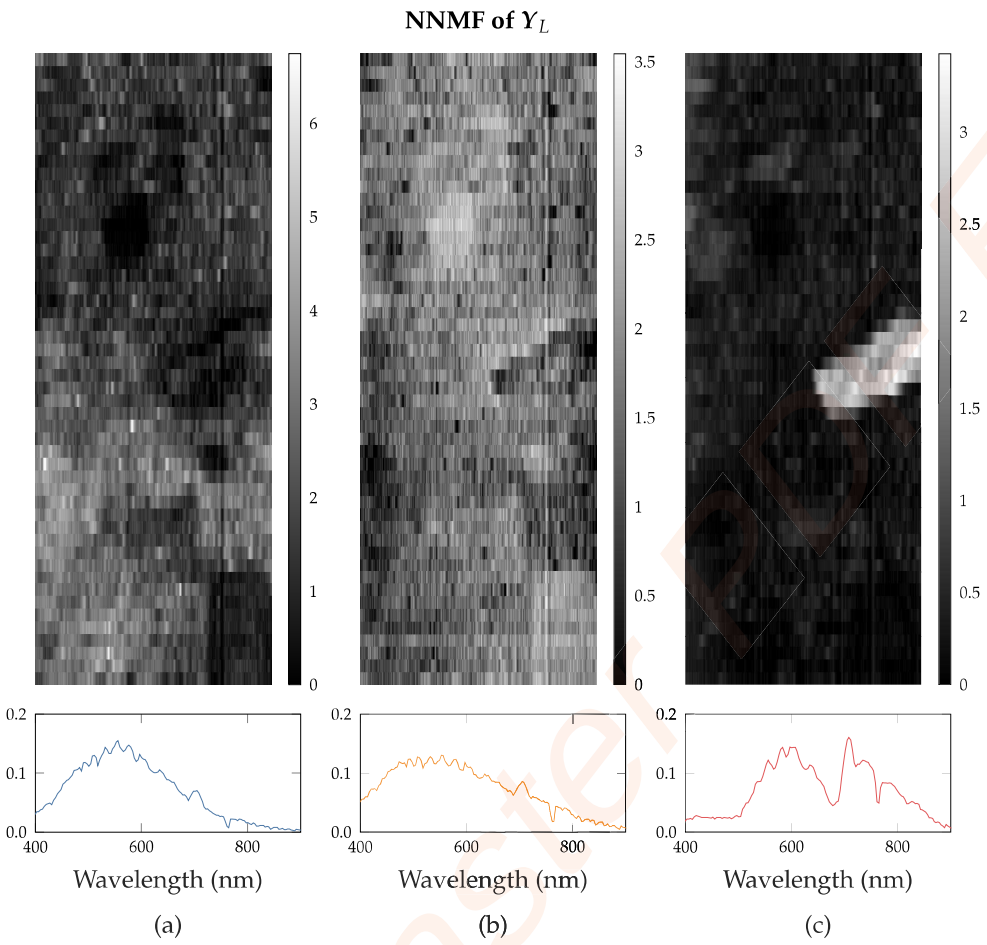


Figure 8: Non-negative Matrix Factorization (NNMF) of low resolution Y_L. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map. In this dataset, raw data is not corrected for solar radiance. Component (a) is strongly influenced by the solar spectrum, showing what looks like intensity of reflected solar spectrum, affected both by the albedo of different materials and in-water path length – i.e. depth , which increases from bottom to top of image, due to ground slope – while (b) seems to pick out the darker rocks. Component (c) is isolating the orange/yellow rock, but the spectrum indicates some influence of chlorophyll, most likely due to the algae covering that rock. Looking at Figure 5 might help understand these components.

minimize this issue by following the convolution, blurring and downsample procedure used in [3], available at [57].

This is not an issue for our field trial dataset, as data comes from two distinct instruments.

Image registration. The work here discussed focuses on the fusion of data with different resolutions, originating from separate sensors. Here we ignore the registration problem,

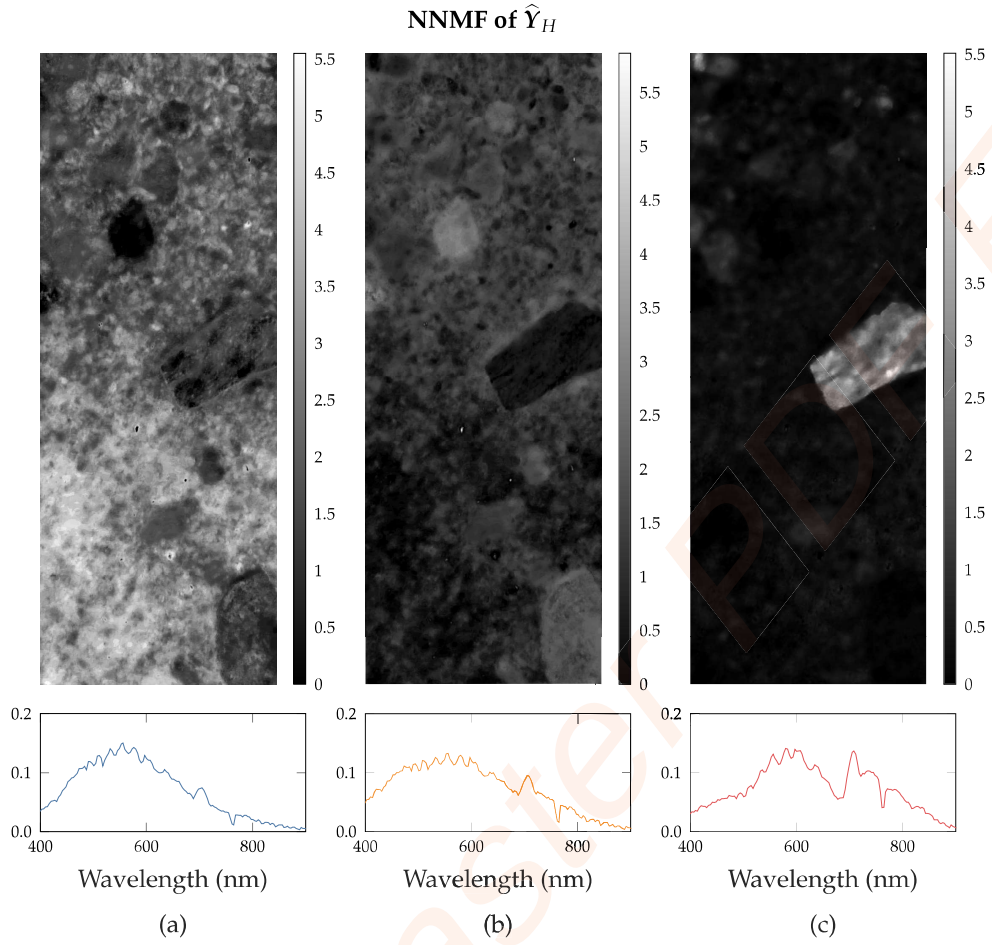


Figure 9: Non-negative Matrix Factorization (NNMF) of high resolution $\widehat{Y}_H$. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map. Notice that all the low resolution artifacts and bad pixels disappear. In this dataset, raw data is not corrected for solar radiance. Component (a) is strongly influenced by the solar spectrum, showing what looks like intensity of reflected solar spectrum, affected both by the albedo of different materials and in-water path length – i.e. depth , which increases from bottom to top of image, due to ground slope – while (b) seems to pick out the darker rocks. Component (c) is isolating the orange/yellow rock, but the spectrum indicates some influence of chlorophyll, most likely due to the algae covering that rock. Looking at Figure 5 might help understand these components.

assuming data is previously aligned and matched. A future development direction would be to integrate data matching and registration as a preprocessing step.

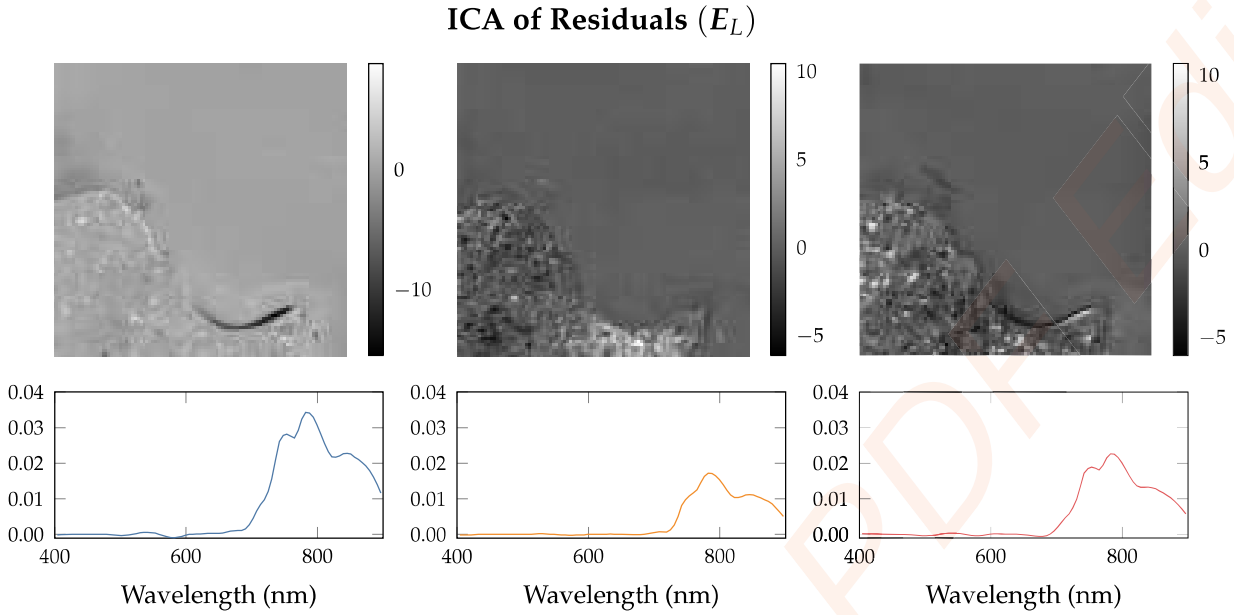


Figure 10: Independent Component Analysis (ICA) of low resolution residuals E_L. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map.

5. Conclusions

In this article we describe a pipeline for enhancing spacial resolution of HS data, taking advantage of co-located RGB data. The method is simple and fast, while giving good quality results.

Furthermore, as the epithet pipeline indicates, the method is composed by a sequence of steps. Here we describe a possible pipeline, where we use a simple projection in the Regression/Estimation step, and expand the number of variables of the RGB data in a certain way. However, the reader may find that for their application, a non-linear regression method and different variations of the high resolution data could work better.

The main contribution of this research is to provide a template for connecting functional blocks, with the aim of fusing multivariate datasets with different resolutions.

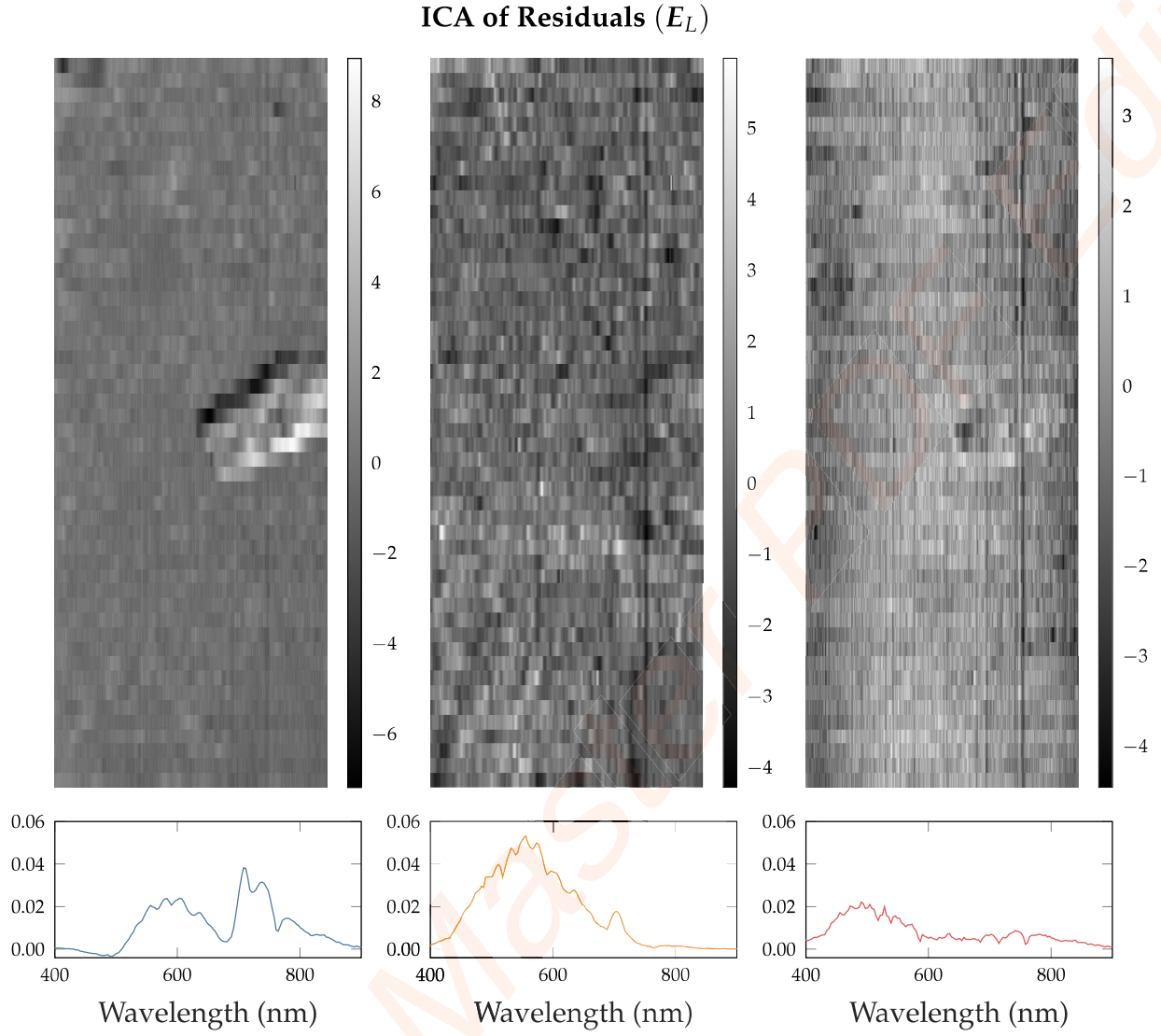


Figure 11: Independent Component Analysis (ICA) of low resolution residuals E_L. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map.

Appendix A. Companion methods

Appendix A.1. Noise Estimation

The method for estimating the noise level per channel is very simple and intuitive, nonetheless the results match the expected instrument performance.

In essence, we check how rough, i.e. non-smooth, each band image is. Gaussian noise will cause sharp peaks very visible when taking the second difference along the horizontal and vertical axes. By looking at the absolute values, we get a direct indication of how noisy a

pixel is.

```
# Diff twice along each direction (vertical and horizontal), for each band image
diff0 = abs(diff(Y_3d, n=2, axis=0))
diff1 = abs(diff(Y_3d, n=2, axis=1))

# Flatten diff results
diff0_flat = reshape(diff0, [diff0.shape[0] * diff0.shape[1], diff0.shape[2]])
diff1_flat = reshape(diff1, [diff1.shape[0] * diff1.shape[1], diff1.shape[2]])
```

To get an overall value per band we can either use the median of all values, horizontal and vertical together:

```
# Vertical and Horizontal all together
diff = concatenate((diff0_flat, diff1_flat), axis=0)
noise = median(diff, axis=0)
```

Or average the vertical and horizontal noises:

```
# Average of Vertical and Horizontal noise levels
median0 = median(diff0_flat, axis=0)
median1 = median(diff1_flat, axis=0)
noise = (median0 + median1) / 2
```

Once the noise level per band is known, see Figure A.12, the inverse of that is used as weights.

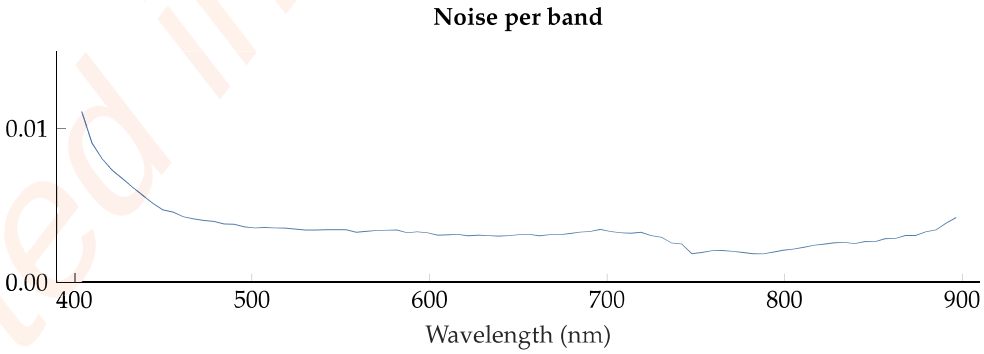


Figure A.12: Noise level per band of low resolution HICO data.

Appendix A.2. Artificial data generation

Here we describe our process to generate RGB and low resolution data, from a high resolution HS reference.

Appendix A.2.1. RGB data

Unlike common practice, instead of picking 3 bands out of the HS data cube we simulate the sensitivity of an RGB sensor. RGB sensors do not have very narrow band-pass filters for Red-Green-Blue wavelengths, so picking one single band to represent each channel would not give us a realistic dataset. Therefore, a weighted sum of different bands extracted from the HS data, with the weights based on the specification sheet of a CMOS RGB sensor, was used for each channel instead, see Figure A.13. This way we ensure,

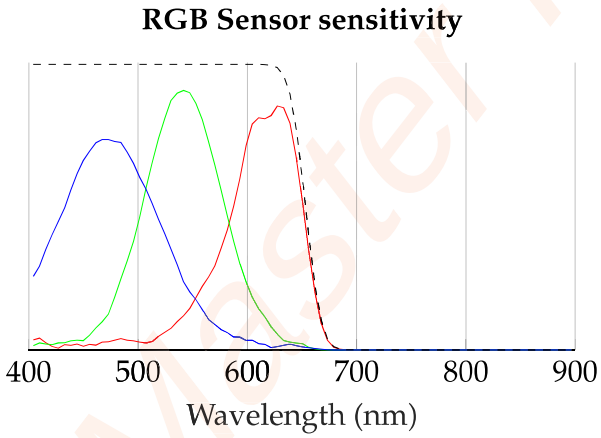


Figure A.13: Weights per band. Weights are designed to simulate an RGB sensor. The black dashed line represents the NIR cut-off filter, usually present in RGB sensors. The vertical scale is unimportant here, we merely want to show the shape of the curves.

through sensible assumptions, that the RGB data are realistic.

Appendix A.2.2. Low resolution HS data

For validation, we generated low resolution data using the same method as the study in [3], available as MATLAB code in [57].

Appendix A.3. Relevant Components

When analysing the residuals before unmixing, the first step is to decompose them through SVD. Then $\boldsymbol{U}$ (Scores), $\boldsymbol{\Sigma}$ (singular values), and $\boldsymbol{V}$ (Loadings) are evaluated by

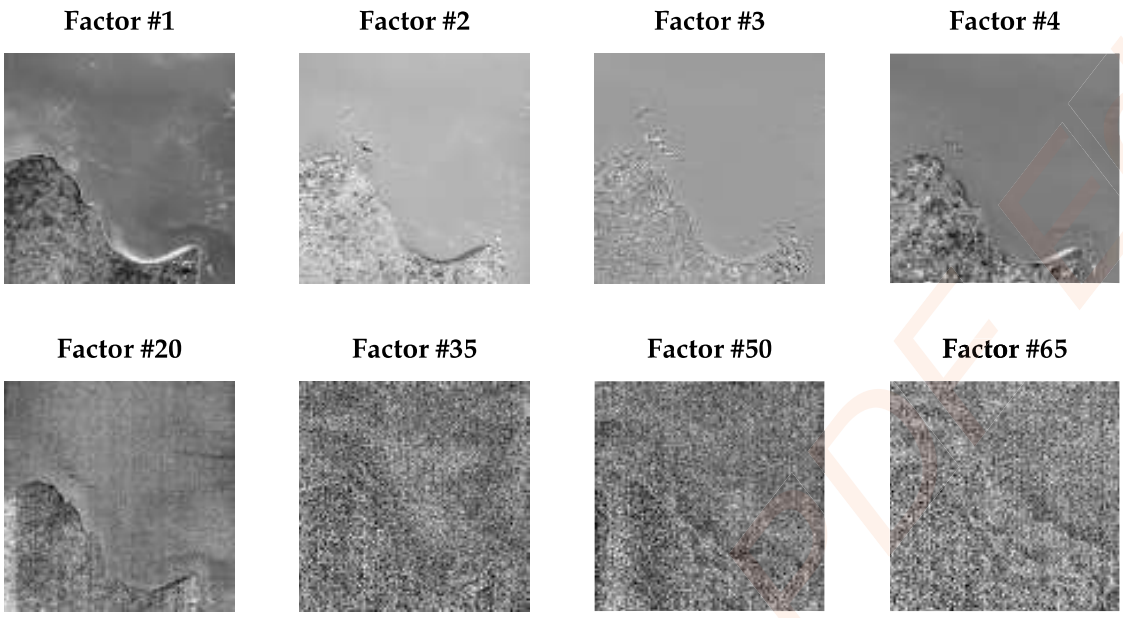


Figure A.14: Spatial Maps (Scores) of a sample of factors from the HICO dataset. Higher factors are clearly more noisy than the lower/earlier.

different methods:

1. the noise level of the spatial distribution maps ($\boldsymbol{U}$)
2. the slope of the scree plot [58] ($\boldsymbol{\Sigma}$)
3. the smoothness of the spectral signatures ($\boldsymbol{V}$)

Noise level of the spatial distribution maps. The same noise per channel routine that was used to find the noise in the raw data is re-used, now on the refolded distribution maps ($\boldsymbol{U}$). Relevant components are expected to have little noise in the spatial domain. Note that $\boldsymbol{U}$ is not scaled with $\boldsymbol{\Sigma}$, so every band has a similar range, and the same threshold can be applied. See Figure A.14

Slope of the scree plot. This is an implementation of an autonomous scree test [58], which is usually a visual inspection test. The scree plot will have an "elbow", which represents the boundary between relevant and non-relevant factors. Through a linear fit, we find the slope of the plateau that corresponds to the noise components, then we start checking lower numbered components until the slope of the linear fit starts to change, indicating we have reached the elbow. See Figure A.15.

```
# Maximum value is always 1,
# this way we can use same threshold values for different data
s /= s.max()
nc = len(s)

# More than half the components are usually noise in remote sensing data,
# so we start from the middle
curr_idx = nc // 2
z = polyfit(x=range(curr_idx, nc), y=s[curr_idx:], deg=1)

while curr_idx > 1:
    curr_idx -= 1
    z_new = polyfit(x=range(curr_idx, nc), y=s[curr_idx:], deg=1)
    fit_chg = abs(z - z_new)
    # If slope changes too much from previous fit, we are past the elbow
    if fit_chg[0] > threshold_fit:
        break
    else:
        z = z_new

p = poly1d(z)
fit_p = p(range(nc))

serr_all = (s - fit_p)**2
serr_fit = serr_all[curr_idx:]

# Relevant components will have large error to fitted line (not in flat region)
vote_slope = serr_all > threshold_error
```

Smoothness of the spectral signatures. Even though the loadings matrix (V) resulting from an SVD of hyperspectral images cannot be directly interpreted as spectral signatures of natural phenomena, which are usually smooth, they are linear combinations of smooth signatures. For sufficiently high spectral resolution sensors, which we assume a hyperspectral camera has, this means that information is smooth in the spectral domain, and noise is not. A smooth spectrum will have small variations in slope. See Figure A.16.

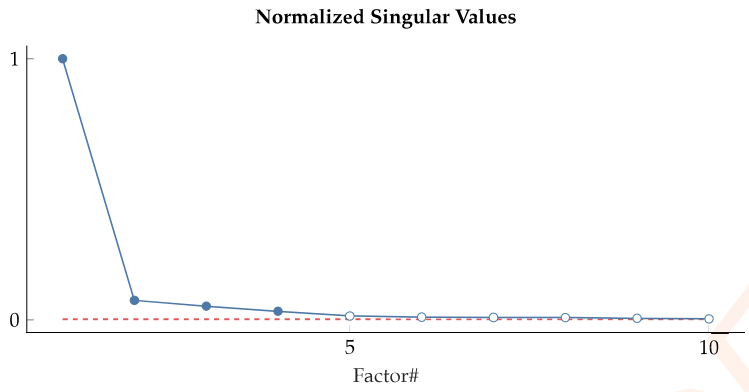


Figure A.15: Scree plot showing only the 10 first components of the low resolution residuals from the HICO dataset. Red dashed line was fitted to the flat section. When the dashed and solid lines diverge we have the relevant factors (marked with filled circles).

```
# Absolute sum of diff along wavelength axis
# large values will show for rough spectra
v_diff = diff(v, n=2, axis=1)
sum_diff = sum(abs(v_diff), axis=1)

# Relevant components have smooth spectra
vote_smooth = sum_diff < threshold_smooth
```

Once we have all the votes from the 3 classifiers, we decide how many factors to keep.

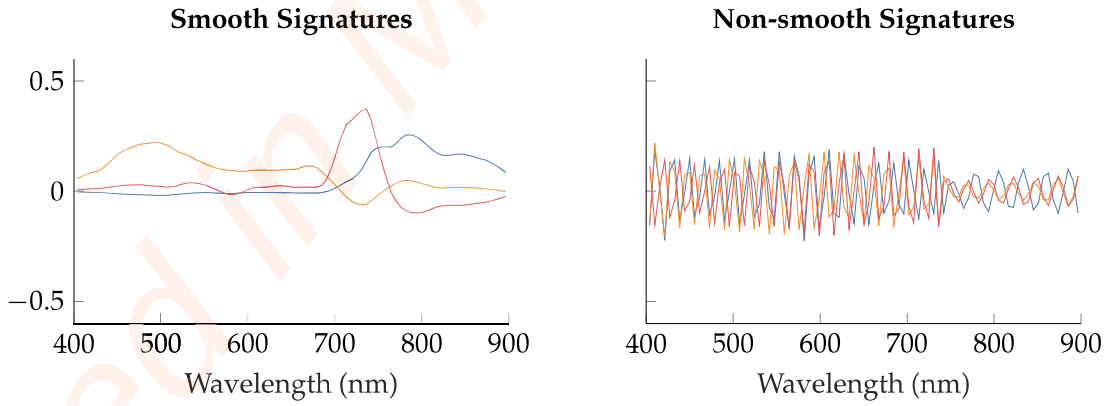


Figure A.16: Example Spectral Signatures (Loadings) split into smooth or non-smooth according to our classifier. The high frequency variations seen in the spectra on the right plot are often correlated with noise. From the HICO dataset.

Figure A.17 shows the result of such voting.

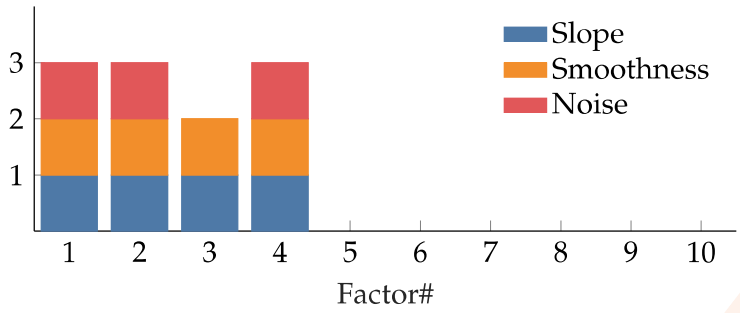


Figure A.17: Votes from each "relevance classifier" for the HICO dataset. Factors with 3 votes are considered relevant. Here, 3 relevant factors should be possible to extract from residuals.

References

[1] C. K. Singh (Ed.), Geospatial Applications for Natural Resources Management, CRC Press, 2018.

[2] F. Sigernes, M. Syrjäsuo, R. Storvold, J. Fortuna, M. E. Grøtte, T. A. Johansen, Do it yourself hyperspectral imager for handheld to airborne operations, Optics Express 26 (2018) 6021.

[3] L. Loncan, L. B. De Almeida, J. M. Bioucas-Dias, X. Briottet, J. Chanussot, N. Dobigeon, S. Fabre, W. Liao, G. A. Licciardi, M. Simoes, J. Y. Tourneret, M. A. Veganzones, G. Vivone, Q. Wei, N. Yokoya, Hyperspectral Pansharpening: A Review, IEEE Geoscience and Remote Sensing Magazine 3 (2015) 27–46.

[4] N. Yokoya, C. Grohnfeldt, J. Chanussot, Hyperspectral and multispectral data fusion: A comparative review of the recent literature, IEEE Geoscience and Remote Sensing Magazine 5 (2017) 29–56.

[5] L. Bungert, D. A. Coomes, M. J. Ehrhardt, J. Rasch, R. Reisenhofer, C.-B. Schönlieb, Blind image fusion for hyperspectral imaging with the directional total variation, Inverse Problems 34 (2018) 044003.

[6] N. Akhtar, F. Shafait, A. Mian, Bayesian sparse representation for hyperspectral image super resolution, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3631–3640.

[7] M. Simoes, J. Bioucas-Dias, L. B. Almeida, J. Chanussot, A convex formulation for hyperspectral image superresolution via subspace-based regularization, IEEE Transactions on Geoscience and Remote Sensing 53 (2015) 3373–3388.

[8] Q. Wei, J. Bioucas-Dias, N. Dobigeon, J. Tourneret, Hyperspectral and multispectral image fusion based on a sparse representation, IEEE Transactions on Geoscience and Remote Sensing 53 (2015) 3658–3668.

[9] Q. Wei, N. Dobigeon, J. Tourneret, Bayesian fusion of multi-band images, IEEE Journal of Selected Topics in Signal Processing 9 (2015) 1117–1127.

[10] Y. Zhang, S. De Backer, P. Scheunders, Noise-resistant wavelet-based bayesian fusion of multispectral and hyperspectral images, IEEE Transactions on Geoscience and Remote Sensing 47 (2009) 3834–3843.

[11] Y. Chang, L. Yan, H. Fang, S. Zhong, Z. Zhang, Weighted low-rank tensor recovery for hyperspectral image restoration, 2017.

[12] R. Dian, L. Fang, S. Li, Hyperspectral image super-resolution via non-local sparse tensor factorization, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3862–3871.

[13] K. Zhang, M. Wang, S. Yang, L. Jiao, Spatial–spectral-graph-regularized low-rank tensor decomposition for multispectral and hyperspectral image fusion, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11 (2018) 1030–1040.

[14] H. Li, L. Jing, Y. Tang, H. Ding, An Improved Pansharpening Method for Misaligned Panchromatic and Multispectral Data, Sensors 18 (2018) 557.

[15] C. I. Kanatsoulis, X. Fu, N. D. Sidiropoulos, W. Ma, Hyperspectral super-resolution: A coupled tensor factorization approach, IEEE Transactions on Signal Processing 66 (2018) 6503–6517.

[16] Y. Xu, Z. Wu, J. Chanussot, Z. Wei, Nonlocal patch tensor sparse representation for hyperspectral image super-resolution, IEEE Transactions on Image Processing 28 (2019) 3034–3047.

[17] R. Dian, S. Li, L. Fang, Learning a low tensor-train rank representation for hyperspectral image super-resolution, IEEE Transactions on Neural Networks and Learning Systems 30 (2019) 2672–2683.

[18] R. Kawakami, Y. Matsushita, J. Wright, M. Ben-Ezra, Y. W. Tai, K. Ikeuchi, High-resolution hyperspectral imaging via matrix factorization, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 2011, pp. 2329–2336.

[19] N. Yokoya, T. Yairi, A. Iwasaki, Coupled Nonnegative Matrix Factorization Unmixing for Hyperspectral and Multispectral Data Fusion, IEEE Transactions on Geoscience and Remote Sensing 50 (2012) 528–537.

[20] N. Akhtar, F. Shafait, A. Mian, Sparse spatio-spectral representation for hyperspectral image super-resolution, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, Springer International Publishing, Cham, 2014, pp. 63–78.

[21] M. Selva, B. Aiazzi, F. Butera, L. Chiarantini, S. Baronti, Hyper-sharpening: A first approach on sim-ga data, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 8 (2015) 3008–3024.

[22] C. Lanaras, E. Baltsavias, K. Schindler, Hyperspectral super-resolution by coupled spectral unmixing, in: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 3586–3594.

[23] H. Kwon, Y. Tai, Rgb-guided hyperspectral image upsampling, in: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 307–315.

[24] Q. Wei, J. Bioucas-Dias, N. Dobigeon, J. Tourneret, M. Chen, S. Godsill, Multiband

image fusion based on spectral unmixing, IEEE Transactions on Geoscience and Remote Sensing 54 (2016) 7236–7249.

[25] W. Dong, F. Fu, G. Shi, X. Cao, J. Wu, G. Li, X. Li, Hyperspectral image super-resolution via non-negative structured sparse representation, IEEE Transactions on Image Processing 25 (2016) 2337–2352.

[26] M. A. Veganzones, M. Simões, G. Licciardi, N. Yokoya, J. M. Bioucas-Dias, J. Chanussot, Hyperspectral super-resolution of locally low rank images from complementary multisource data, IEEE Transactions on Image Processing 25 (2016) 274–288.

[27] K. Zhang, M. Wang, S. Yang, Multispectral and hyperspectral image fusion based on group spectral embedding and low-rank factorization, IEEE Transactions on Geoscience and Remote Sensing 55 (2017) 1363–1371.

[28] C. Yi, Y. Zhao, J. C. Chan, Hyperspectral image super-resolution based on spatial and spectral correlation fusion, IEEE Transactions on Geoscience and Remote Sensing 56 (2018) 4165–4177.

[29] X. Han, B. Shi, Y. Zheng, Self-similarity constrained sparse representation for hyperspectral image super-resolution, IEEE Transactions on Image Processing 27 (2018) 5625–5637.

[30] L. Zhang, W. Wei, C. Bai, Y. Gao, Y. Zhang, Exploiting clustering manifold structure for hyperspectral imagery super-resolution, IEEE Transactions on Image Processing 27 (2018) 5969–5982.

[31] Y. Fu, Y. Zheng, H. Huang, I. Sato, Y. Sato, Hyperspectral image super-resolution with a mosaic rgb image, IEEE Transactions on Image Processing 27 (2018) 5539–5552.

[32] Z. Pan, H. Shen, Multispectral image super-resolution via rgb image fusion and radiometric calibration, IEEE Transactions on Image Processing 28 (2019) 1783–1797.

[33] R. Dian, S. Li, Hyperspectral image super-resolution via subspace-based low tensor multi-rank regularization, IEEE Transactions on Image Processing 28 (2019) 5135–5146.

[34] C. Wang, Y. Liu, X. Bai, W. Tang, P. Lei, J. Zhou, Deep residual convolutional neural network for hyperspectral image super-resolution, in: Y. Zhao, X. Kong, D. Taubman (Eds.), Image and Graphics, Springer International Publishing, Cham, 2017, pp. 370–380.

[35] X. Han, B. Shi, Y. Zheng, Ssf-cnn: Spatial and spectral fusion with cnn for hyperspectral image super-resolution, in: 2018 25th IEEE International Conference on Image Processing (ICIP), pp. 2506–2510.

[36] R. Dian, S. Li, A. Guo, L. Fang, Deep hyperspectral image sharpening, IEEE Transactions on Neural Networks and Learning Systems 29 (2018) 5345–5355.

[37] Y. Chang, L. Yan, H. Fang, S. Zhong, W. Liao, Hsi-denet: Hyperspectral image restoration via convolutional neural network, IEEE Transactions on Geoscience and Remote Sensing 57 (2019) 667–682.

[38] Y. Qu, H. Qi, C. Kwan, Unsupervised sparse dirichlet-net for hyperspectral image super-resolution, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2511–2520.

[39] O. Sidorov, J. Y. Hardeberg, Deep hyperspectral prior: Denoising, inpainting, super-resolution, 2019.

[40] Q. Xie, M. Zhou, Q. Zhao, D. Meng, W. Zuo, Z. Xu, Multispectral and hyperspectral image fusion by ms/hs fusion net, 2019.

[41] X.-H. Han, Y. Zheng, Y.-W. Chen, Multi-level and multi-scale spatial and spectral fusion cnn for hyperspectral image super-resolution, in: The IEEE International Conference on Computer Vision (ICCV) Workshops.

[42] Y. Zhou, A. Rangarajan, P. D. Gader, An integrated approach to registration and fusion of hyperspectral and multispectral images, IEEE Transactions on Geoscience and Remote Sensing (2019) 1–14.

[43] W. Wang, W. Zeng, Y. Huang, X. Ding, J. Paisley, Deep blind hyperspectral image fusion, in: The IEEE International Conference on Computer Vision (ICCV).

[44] NASA (@nasa) on Unsplash, Water, nature, ocean and reef, https://unsplash.com/photos/6-jTZysYY_U, 2019. Accessed: 2019-08-17.

[45] J. Fortuna, T. A. Johansen, A lightweight payload for hyperspectral remote sensing using small uavs, in: 2018 9th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), pp. 1–5.

[46] C. Bedia, À. Sierra, R. Tauler, Application of chemometric methods to the analysis of multimodal chemical images of biological tissues, Analytical and Bioanalytical Chemistry (2020).

[47] S. Piqueras, C. Bedia, C. Beleites, C. Krafft, J. Popp, M. Maeder, R. Tauler, A. de Juan, Handling different spatial resolutions in image fusion by multivariate curve resolution-alternating least squares for incomplete image multisets, Analytical Chemistry 90 (2018) 6757–6765.

[48] J. Fortuna, H. Martens, Multivariate data modelling for de-shadowing of airborne hyperspectral imaging, Journal of Spectral Imaging 6 (2017).

[49] H. Martens, T. Naes, Multivariate calibration, John Wiley & Sons, 1992.

[50] G. H. Golub, C. Reinsch, Singular value decomposition and least squares solutions, in: Linear Algebra, Springer, 1971, pp. 134–151.

[51] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (1999) 788–791.

[52] A. de Juan, R. Tauler, Multivariate Curve Resolution (MCR) from 2000: Progress in concepts and applications, Critical Reviews in Analytical Chemistry 36 (2006) 163–176.

[53] A. Hyvärinen, E. Oja, Independent component analysis: algorithms and applications, Neural networks 13 (2000) 411–430.

[54] I. T. Jolliffe, Principal Component Analysis, Springer Series in Statistics, Springer-Verlag, New York, 2 edition, 2002.

[55] Oregon State University, HICO - Hyperspectral Imager for the Coastal Ocean, http://hico.coas.oregonstate.edu/, 2019. Accessed: 2019-08-08.

[56] A. Hyvärinen, E. Oja, Independent component analysis: algorithms and applications, Neural Networks 13 (2000) 411–430.

[57] Open Remote Sensing, Hyperspectral Pansharpening : A review, https://openremotesensing.net/knowledgebase/hyperspectral-pansharpening-a-review/, 2015. Accessed: 2019-04-10.

[58] R. B. D'agostino Sr, H. K. Russell, Scree test, in: Wiley StatsRef: Statistics Reference Online, American Cancer Society, 2014.

- Hyperspectral instruments are widely used for remote sensing applications
- Push-broom hyperspectral has low spatial resolution when compared to color cameras
- Observations of the same target by hyperspectral and color sensors can be merged
- Multivariate calibration can be applied to combine data from the two sensors

Multivariate Image Fusion: A Pipeline For Hyperspectral Data Enhancement

João Fortuna[a,b,1,*], Harald Martens[a,c], Tor Arne Johansen[a,b]

[a]*Norwegian University of Science and Technology (NTNU), Department of Engineering Cybernetics, Trondheim, Norway*
[b]*Centre for Autonomous Marine Operations and Systems (NTNU AMOS), Trondheim, Norway*
[c]*Idletechs AS, Trondheim, Norway*

Abstract

Hyperspectral cameras provide high spectral resolution data, but their usual low spatial resolution when compared to color (RGB) instruments is still a limitation for more detailed studies. This article presents a simple yet powerful method for fusing co-registered high spatial and low spectral resolution image data – e.g. RGB – with low spatial and high spectral resolution data – Hyperspectral. The proposed method exploits the overlap in observed phenomena by the two cameras to create a model through least square projections. This yields two images: 1) A high-resolution image spatially correlated with the input RGB image but with more spectral information than just the 3 RGB bands. 2) A low-resolution image showing the spectral information what is spatially uncorrelated with the RGB image. We show results for semi-artificial benchmark datasets and a real-world application. Performance metrics indicate the method is well suited for data enhancement.

Keywords: Hyperspectral, Data Fusion, Pansharpening, Super Resolution

1. Introduction

Information on distribution and abundance of natural resources is important for science, education, policy making and management alike [1]. Hyperspectral (HS) instruments provide a richness of data that enables classification and detection of such resources

*Corresponding author
Email address: joao.fortuna@ntnu.no (João Fortuna)

[1]This work was supported by the Norwegian Research Council (grant no. 223254) through the Centre of Autonomous Marine Operations and Systems (NTNU AMOS) at the Norwegian University of Science and Technology, the MASSIVE project (grant no. 270959), as well as the Norwegian Space Center.

through passive and non destructive measurements. Furthermore, they can scan large extents of ground in a short time period, making them well suited tools for air- and space-borne remote sensing. Low-cost hyperspectral systems recently developed [2] make this technology more accessible to research groups all over the world. While such low-cost systems are not expected to produce the same high quality data as more expensive equipment, they may be well suited for certain applications.

Using multiple sensors with different capabilities, often creates a clearer picture of the environment when compared to a single sensor scenario. However, while our brains are good at fusing information from different sources, some work is required if the process is to be automated. Multivariate calibration, a term coined in the field of chemometrics, refers to the development of models to explain certain properties of interest by combining different variables from multi-channel sensor measurements.

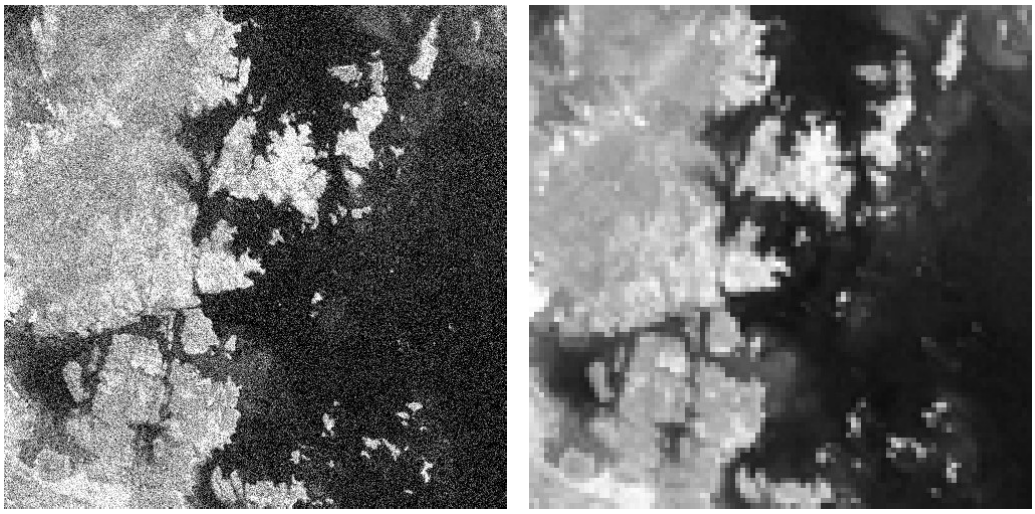
Fusing images from sensors with different spectral and spatial properties to generate a single, improved data product is a known and studied problem [3, 4]. Hyperspectral image super-resolution fusion methods can be grouped into 4 categories: Bayesian based approaches [5–11]; Tensor based approaches [12–17]; Matrix factorization based approaches [18–33]; and Deep Learning based approaches [34–41].

The mentioned methods assume that the images to be fused are co-located (registered). However, more recent methods drop that assumption and achieve simultaneous registration and super-resolution [42, 43].

1.1. There is no such thing as a free lunch, or photons

Even though **number of pixels** is a different concept from **spatial resolution**, they are tightly coupled. Say we have two monochrome focal plane array (FPA) cameras, A and B, on a satellite and both image the same area on the ground, see Figure 1. If A has 5 times the amount of pixels in both axis, then the spatial resolution – the ability to differentiate between two close objects, or in this case the size of one pixel on the ground – will be 5 times better than the one of camera B. Here we assume all other properties of the cameras, such as optics, and image capture to be the same and that they do not limit the resolution, i.e. the sensor is the bottleneck.

Higher resolution has the obvious benefit of allowing finer details to be seen, however,



(a) High resolution (500×500 px) (b) Low resolution (100×100 px)

Figure 1: Example satellite images with simulated different resolutions. Sensors have same area but different number of pixels (different pixel size and density). The higher number of pixels in (a) means less light (fewer photons) per pixel, hence more noise. Photo by NASA on Unsplash [44].

increasing the number of pixels is not always possible, particularly as we increase the number of spectral bands. To understand why the number of bands affects the spatial resolution, we need to think of light reaching a camera as a stream of a finite number of photons. These particles need to travel through the optical elements of the camera, then they are distributed by all the pixels in the sensor, where they are transformed into an electrical current that is finally converted into digital data. Higher pixel density means less photons per pixel, as we divide the same finite amount of photons by a higher number of pixels, and fewer photons per pixel leads to a weaker signal (low SNR). When the signal is fainter, the noise contribution becomes apparent if we increase the sensor gain, compare (a) and (b) in Figure 1. If we now try to sort the photons into many spectral bands, they become even scarcer and we need to compromise on the number of spatial pixels in order to still have a usable signal.

Throughout this article we will mention high and low resolution data, in this scope we mean both the number of pixels and ability to resolve a smaller object in the image. Higher resolution data will have more pixels and conversely, lower resolution, fewer pixels.

Panchromatic images contain information from a broad spectrum in a single band, hence

can more easily have a high spatial resolution. Pansharpening methods were initially developed in the mid 1980s for air- and space-borne multispectral imagers with low resolution that could be improved with high resolution panchromatic images. With the ever growing availability of hyperspectral instruments, some of those methods were adapted to hyperspectral data and others developed anew. Several methods to achieve such sharpening are described in the literature [3].

Thanks to advances in sensor technology, we now have spatially high resolution color cameras (Red-Green-Blue – RGB) and even some multispectral cameras (with few, but more than 3, bands), which we can use instead of panchromatic when enhancing hyperspectral data. The advantages of using colour cameras are clear: Even with only 3 bands we have multivariate – as opposed to univariate – spectral data in high spatial resolution. Such multivariate data gives much better selectivity, as it adds color information to the simple measurement of light intensity provided by panchromatic sensors. This is important in the present setting, where a spatially high-resolution RGB camera is combined with a spatially low-resolution hyperspectral camera: The higher number of bands with high spatial resolution, the more we can improve the spatial resolution of hyperspectral data, assuming that both cameras have recorded the same spatial scene and therefore can be correlated.

Some of the most commonly used pansharpening methods are useful when the intended goal is to produce a high resolution RGB image from multi-/hyperspectral data, having only a high resolution monochromatic image. However, those enhancements are not visible when the goal is, for example, to find high resolution estimated abundance maps of geological or biological resources of interest.

1.2. Motivation

For the past years, our research group has been working on a lightweight hyperspectral imaging system for unmanned aerial systems (UAS) [2][45]. Because of weight, cost and complexity limitations, and operating conditions, the spatial resolution has been the most limiting factor when it comes to generating high quality data products. RGB cameras are very often already a part of the payload carried by such unmanned systems, and if not, they are a simple addition.

Taking that into consideration, trying to improve HS spatial resolution with RGB data was set as the goal for the present research.

1.3. Paper contribution

In this paper we describe a generic framework for multivariate image fusion, building on the ideas of pansharpening while trying to also enhance the output for further processing, instead of just visual representation. Hence, the purpose of the present methodology is to use the high spatial resolution of the RGB measurements to yield an equally high resolution representation of the low resolution HS measurements. Other studies [46], [47] have pursued somewhat related approaches, nevertheless, the method proposed here is to the authors' knowledge, a new development. It is fast, when compared to other methods in the literature [3], and requires very little knowledge of calibration parameters or relationship between the two datasets to be fused, only assuming that they have been spatially registered beforehand. In summary, the present method combines the input, consisting of a low-resolution multi-channel HSI image and a high-resolution 3-channel RGB image of the same scene, into two output images: 1) A high-resolution multi-channel image showing what is spatially correlated to the RGB image, and 2) A low-resolution multi-channel image showing what is spatially uncorrelated to the RGB image. The following section describes each step of the method. We then show some results of applying the method to both artificially degraded real data, and real low resolution data. To conclude, a discussion on possible improvements and future work.

2. Method - Multivariate Image Fusion (MVIF)

In this section we present a method for fusing RGB and HS data in order to get a data product that takes advantage of the relative strengths of both, Figure 2. Such fusion is possible because HS data is in most real world applications typically very rank deficient — the number of HS wavelength bands is much higher than the number of statistically independent spectral variation types in the image. That means it is possible to learn all relevant patterns of variation in HS and replicate them through a low rank, but high spatial resolution, approximation based on RGB.

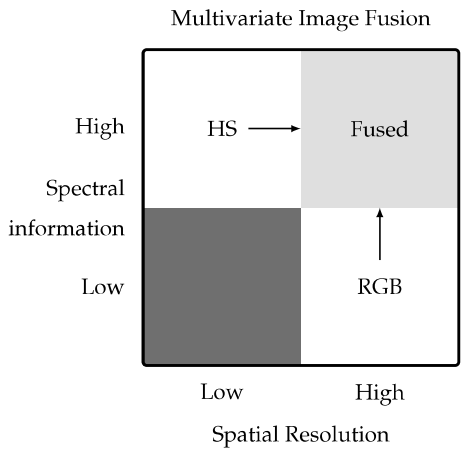


Figure 2: Multivariate Image Fusion exploits the relative strengths of RGB and Hyperspectral data.

2.1. Data Model

Before continuing, it is useful to write down the data model we will be working with. We take a similar approach as [48], based on the hypothetical model:

$$\boldsymbol{Y} = \boldsymbol{C}\boldsymbol{S}^{\mathsf{T}} + \boldsymbol{D}\boldsymbol{Z}^{\mathsf{T}} + \boldsymbol{F} \tag{1}$$

In this model we assume that data cubes are unfolded, so all elements are 2D matrices.
$\boldsymbol{Y}$ is a high spectral resolution (hyperspectral, here called "full-spectral") image dataset, with high or low spatial resolution, depending on the context.
$\boldsymbol{C}$ is the RGB data. The product $\boldsymbol{C}\boldsymbol{S}^{\mathsf{T}}$ contains the known spatial pattern from the RGB camera ($\boldsymbol{C}$), based on initially unknown, but estimated full-spectral ($\boldsymbol{S}^{\mathsf{T}}$) information.
In analogy to the previous element, $\boldsymbol{D}\boldsymbol{Z}^{\mathsf{T}}$ contains spatial ($\boldsymbol{D}$) and spectral ($\boldsymbol{Z}^{\mathsf{T}}$) information, but now from phenomena that are not seen by the RGB camera. Initially unknown, ($\boldsymbol{D}$) and spectral ($\boldsymbol{Z}^{\mathsf{T}}$) have to be estimated.
$\boldsymbol{F}$ will, ideally, contain only noise.
When dealing with remote sensing spectral data, it is helpful to think of the total signal in each pixel of $\boldsymbol{Y}$ as a sum of contributions from all the phenomena that were observed in that single pixel. These contributions have 2 properties: concentration/abundance $\boldsymbol{C}$ and $\boldsymbol{D}$ and spectral signature $\boldsymbol{S}$ or $\boldsymbol{Z}$. When we consider all the pixels in $\boldsymbol{Y}$, the concentrations become spatial distribution maps of each of those phenomena. Adding all the spectral signatures – $\boldsymbol{S}$ and $\boldsymbol{Z}$ – weighted according to their respective concentration per pixel – $\boldsymbol{C}$

and $\boldsymbol{D}$ – yields the signal in $\boldsymbol{Y}$, aside from noise – $\boldsymbol{F}$:

$$\boldsymbol{Y} = [\boldsymbol{C}, \boldsymbol{D}]\,[\boldsymbol{S}, \boldsymbol{Z}]^{\mathsf{T}} + \boldsymbol{F} \tag{2}$$

Even though the number of bands (spectral resolution) can be of several tens or even hundreds, hyperspectral data is typically rank deficient, which means we can obtain a much lower dimension representation with less noise, while still keeping the relevant information. On the other hand, RGB data usually has a full rank of 3 in the spectral domain. When the spectral range of both instruments is overlapping, they observe the same phenomena and we can obtain a low rank representation of HS using RGB data.
If we use RGB data as is – with 3 bands – we are limited to a 3 dimension low rank representation, and while this may be enough for some datasets, it will prove inadequate for more complex scenes. This is a limitation of using a linear projection method as we have done here, other methods may not face this problem. Fortunately, it is possible to artificially expand the number of high resolution bands by appending the result of non-linear operations on the original RGB data, thus increasing the rank of the high resolution data. Such operations are for example: interaction terms (product of different bands) or square terms:

$$\boldsymbol{C}_{RGB} = [\boldsymbol{C}_R, \boldsymbol{C}_G, \boldsymbol{C}_B] \tag{3}$$

$$\boldsymbol{C}_{int} = [\boldsymbol{C}_R \circ \boldsymbol{C}_G, \boldsymbol{C}_R \circ \boldsymbol{C}_B, \boldsymbol{C}_G \circ \boldsymbol{C}_B] \tag{4}$$

$$\boldsymbol{C}_{sqr} = [\boldsymbol{C}_R \circ \boldsymbol{C}_R, \boldsymbol{C}_G \circ \boldsymbol{C}_G, \boldsymbol{C}_B \circ \boldsymbol{C}_B] \tag{5}$$

$$\boldsymbol{C}_{ext} = \left[\boldsymbol{C}_{RGB}, \boldsymbol{C}_{int}, \boldsymbol{C}_{sqr}\right] \tag{6}$$

If we consider $\boldsymbol{C}_{RGB}$ – Eq. 3 – to be the original RGB-only data, unfolded, where each column represents a color channel, then we can define a matrix of first-degree interactions, $\boldsymbol{C}_{int}$, and another of square terms, $\boldsymbol{C}_{sqr}$, respectively by element-wise multiplying each band by another, Eq. 4, or by itself, Eq. 5. The operator $\circ$ represents the element-wise multiplication, also known as Hadamard product. When composing the $\boldsymbol{C}$ matrix to input to the algorithm, we could use $\boldsymbol{C}_{ext}$ if we wanted to include interaction and square terms. Fundamentally, interaction and square terms do not add new information, however by

providing these non-linear terms to the linear algorithm, it allows it to find non-linear spectral variations, which are expected to exist. This is analogous to how different wavelength channels have different non-linear relations to the chemical sample composition in NIR multichannel reflectance measurements. These unknown but different non-linear relations may be regarded as a special type of unknown interference. Using the pragmatic but incorrect $\log(1/R)$ transform allows linear multivariate calibration modelling, e.g. by PLSR to utilize the additional subspace dimensions, spanned by the channels' unknown differences in non-linearity, to pick up and correct for these unknown interference, as described in [49].

For our present RGB data we do not know the detailed camera properties. In addition, the light signal is affected by the atmospheric absorbance and light scattering effects in the water phase that the photons have to go through, on their way from the light source, the Sun, via the bottom object and back to the camera. We do not know the ideal mathematical transform from chemical and physical properties of the objects on the bottom, to the RGB signal of the camera, but the transform $\log(1/R)$ is probably too simplistic. However, by adding new "wavelength channels" by non-linear combinations, e.g. interactions and square terms, of the original RGB channels, the linear multivariate calibration has a better chance of finding a subspace that spans both the chemical and physical signal variations and their non-linearities.

Later, when we discuss the performance of our proposed method, we show how different combinations of non-linear terms affected it.

2.2. Notation

Throughout the paper we will use the following notation:

- Unfolded HS data cube with k_H bands and low spatial resolution $n_L^h \times n_L^w$ — $\boldsymbol{Y}_L \in \mathbb{R}^{n_L \times k_H}$
- Unfolded RGB data (and appended artificially generated terms) with k_L bands and high spatial resolution $n_H^h \times n_H^w$ — $\boldsymbol{C}_H \in \mathbb{R}^{n_H \times k_L}$
- Enhanced HS data with high spectral and spatial resolutions — $\widehat{\boldsymbol{Y}}_H \in \mathbb{R}^{n_H \times k_H}$

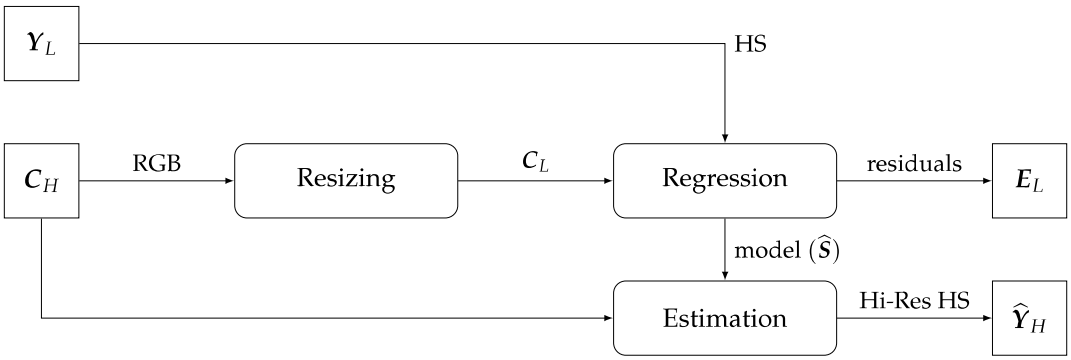


Figure 3: Pipeline overview diagram.

Where $n_L = n_L^h \times n_L^w$ and $n_H = n_H^h \times n_H^w$ are respectively the total number of spatial pixels – height (h) and width (w) – in low and high resolution data.

2.3. Method overview

The algorithm can be summarized in the following steps, also visible in Figure 3:

1. **Resizing** (shrinking) high-resolution RGB data to low-resolution HS size, with image registration, to ensure that the pixels in both images represent the same ground positions.
2. Noise weighted modelling to estimate HS from low-resolution RGB by regression over the low resolution pixels, through **Regression**. Lack-of-fit HS residuals are kept for further analysis of spectral patterns not seen in RGB.
3. **Estimation** of HS using high-resolution RGB.

2.4. Resizing

As mentioned before, $\boldsymbol{Y}_L$ and $\boldsymbol{C}_H$ have different image dimensions, so we start by shrinking $\boldsymbol{C}_H$ for it to coincide $\boldsymbol{C}_L$. Resizing should take into account the properties of the HS instrument that resulted in such low resolution – i.e. if sampling frequency is low but exposure time is also low, resulting in a subsampled target, then we should resample the high resolution RGB data. If on the other hand, the exposure time is long and the target (ground) is fully sampled/observed, but the information is mixed/convolved due to motion blur in each pixel, then we should apply a similar convolution to mix the RGB pixels. This way we ensure that the ground contributions are similarly represented between the HS and RGB data.

2.5. Noise-balancing wavelength weights

Hyperspectral instruments have varying levels of noise for each band. The proposed methodology involves least-squares based estimation of parameters in a reduced rank regression model. For such methods it is important to balance the noise level of the different wavelength channels. For the most common pushbroom slit-grating design the noise is generally worse as we move away from the center of the sensor and the operation range of the (electro-)optical components. In VIS-NIR instruments, usually with CMOS sensors, performance degrades quickly for bands below 400nm and above 900nm, but even inside that range, the noise level varies from channel to channel. Knowing how the noise varies improves the modeling performance, by down-weighing noisy bands we reduce the risk of over fitting noise. We estimated the noise according to the method described in Appendix A.1.

2.6. Regression and Estimation

Once pre-processing is done, we can proceed to the core of our method, the regression step. Here we estimate $\boldsymbol{S}$, knowing $\boldsymbol{Y}$ and $\boldsymbol{C}$. Generically speaking, we establish a projection model:

$$\boldsymbol{Y} = \boldsymbol{C}\boldsymbol{S}^{\mathsf{T}} + \boldsymbol{E} \tag{7}$$

Then we can apply the previous equation to our data:

$$\boldsymbol{Y}_L = \boldsymbol{C}_L\boldsymbol{S}^{\mathsf{T}} + \boldsymbol{E}_L \tag{8}$$

$$\widehat{\boldsymbol{S}}^{\mathsf{T}} = (\boldsymbol{C}_L^{\mathsf{T}}\boldsymbol{C}_L)^{-1}\boldsymbol{C}_L^{\mathsf{T}}\boldsymbol{Y}_L \tag{9}$$

$$\widehat{\boldsymbol{Y}}_L = \boldsymbol{C}_L\widehat{\boldsymbol{S}}^{\mathsf{T}} \tag{10}$$

$$\boldsymbol{E}_L = \boldsymbol{Y}_L - \widehat{\boldsymbol{Y}}_L \tag{11}$$

The matrix $\widehat{\boldsymbol{S}}$ contains the estimate of a dictionary that translates the variations in $\boldsymbol{C}_L$ into variations in $\boldsymbol{Y}_L$. Furthermore, we can now use $\boldsymbol{C}_H$ with that same dictionary and compute:

$$\widehat{\boldsymbol{Y}}_H = \boldsymbol{C}_H\widehat{\boldsymbol{S}}^{\mathsf{T}} \tag{12}$$

which gives a high spatial resolution estimation of $\boldsymbol{Y}$.

2.7. Low resolution residuals analysis

When estimating $\widehat{\boldsymbol{S}}$ and $\widehat{\boldsymbol{Y}}_L$, we are left with unmodelled low-resolution residuals – $\boldsymbol{E}_L$. These can be analysed to give us some insight into what could not be enhanced to higher resolution – systematic information not captured by the model – and estimate how much of it was random independent noise. In order to do that, we need to further decompose $\boldsymbol{E}_L$, through some bilinear matrix decomposition techniques, according to the model:

$$\boldsymbol{E}_L = \boldsymbol{D}_L \boldsymbol{Z}^\intercal + \boldsymbol{F}_L \tag{13}$$

$$\boldsymbol{D}_L \in \mathbb{R}^{n_L \times A} \tag{14}$$

$$\boldsymbol{Z} \in \mathbb{R}^{k_H \times A} \tag{15}$$

where A is the number of factors, or components extracted.

Referring back to our overall model in Eq. 1, then $\boldsymbol{D}_L$ and $\boldsymbol{Z}$ contain low-resolution spatial and respective spectral information of phenomena that are not measured by the RGB camera. $\boldsymbol{F}$ contains unmodeled noise, in low-resolution.

2.7.1. Matrix decomposition

Matrix decomposition or factorization, also called unmixing in the context of spectral data, is a family of methods that split a matrix in a product of other matrices. For hyperspectral data, those resulting matrices usually correspond to some type of spectral signatures and respective spatial distribution and/or concentration.

A simple yet useful factorization method is Singular Value Decomposition (SVD) [50][49]. However, a property of the resulting spectral features is that they are orthonormal, hence not directly representative of bio/geo/chemical spectral signatures.

On the other hand, with Non-Negative Matrix Factorization (NNMF) [51], Multivariate Curve Resolution (MCR) [52] or Independent Component Analysis (ICA) [53], the spectral components (loadings) are often related to actual spectral signatures of phenomena seen in the captured scene. This comes at the expense of more complex computation.

Hyperspectral data is notoriously rank deficient, meaning that factorization methods will model a limited number of meaningful components and many noise components.

For the implementation of the MVIF pipeline here described we first use SVD to estimate

the number of non-noise components (A) in the residuals, then we can use one of the more complex methods knowing how many components to expect.

There is not a consensus among specialists regarding which is the best method to select the appropriate number of relevant components when using SVD [54], furthermore many of them require visual inspection of plots. We propose a solution based on a voting system: three methods evaluate different metrics and vote on whether a component is relevant or not. If a component gets the all votes, it is deemed relevant. Find more details about this method in Appendix Appendix A.3.

Once we have all the votes from the 3 classifiers, we decide how many factors to keep – A. Then it is simply a matter of running the unmixing method of our choice, to obtain the factorization as in Equation 13.

$$\boldsymbol{E}_L \xrightarrow[\text{decomp}]{\text{matrix}} \widehat{\boldsymbol{D}}_{L,A}, \widehat{\boldsymbol{Z}}_A \tag{16}$$

$$\widehat{\boldsymbol{E}}_{L,A} = \widehat{\boldsymbol{D}}_{L,A}\widehat{\boldsymbol{Z}}_A^{\mathsf{T}} \tag{17}$$

$$\boldsymbol{F}_{L,A} = \boldsymbol{E}_L - \widehat{\boldsymbol{E}}_{L,A} \tag{18}$$

3. Results

In this section we show and analyse the results of applying the presented method to two distinct datasets. In addition, we also compare metrics for a third benchmark dataset. First, we generate low resolution hyperspectral data by degrading a high resolution hyperspectral data cube. RGB data is also extracted from the high resolution HS data. As we have a high resolution reference, we can quantify the performance of the algorithm using the performance metrics described in [3].

Second, we use a sample of data from a UAV field campaigns, for which this method was conceived for. RGB comes from a separate camera. Since there is no high resolution reference, performance can only be evaluated visually.

Finally, the benchmark dataset is just briefly analysed in order to compare the performance of MVIF to that of another method from literature on the same dataset.

(a) RGB representation of $\boldsymbol{Y}_H$, which in this case is also $\boldsymbol{C}_H$.

(b) RGB representation of $\boldsymbol{Y}_L$.

Figure 4: RGB rendering of HICO scene used during tests. The procedure described in Appendix A.2 was used to create both the high (500×500 px) and low resolution (100×100 px) images. Some color adjustment was applied to make the images more aesthetically pleasing.

3.1. Data

3.1.1. Control Dataset

We used data from the HICO instrument, available at [55]. HICO (Hyperspectral Imager for the Coastal Ocean) was a hyperspectral imager that was installed on the International Space Station (ISS) and captured data from 2009 to 2014. In terms of specifications, it has 87 bands (400-900 nm), cross-track resolution of 500 pixels and ground sample distance (GSD) of 90 m.

Both the RGB representation of the HS data and the reference RGB image were, for this dataset, extracted from the HS data – the usual procedure for benchmarking these types of algorithms. See Appendix A.2 for more details.

We used a sample of HICO data at full resolution as the high-resolution HS reference – dataset ID H2011145084342. From this dataset we extracted a 500×500 region of interest, see Figure 4. The low resolution HS data had 100×100 spatial pixels and 87 bands.

3.1.2. Test/Field Dataset

We also include the results of applying our method to data obtained in a field trial with drone mounted HS and RGB cameras, see Figure 5. This flight was conducted in Hopavågen, Norway, in March 2018. More details on the data capture and the experiment are available in [45].

In this dataset, the loss of spatial resolution was due to sub-optimal flight conditions, and instrument design limitations. The across-track resolution (here seen as the horizontal axis) is equivalent in both RGB and HS, but the along-track direction (vertical axis) is much lower for the HS camera. In fact, for each HS pixel, there are 30 RGB pixels.

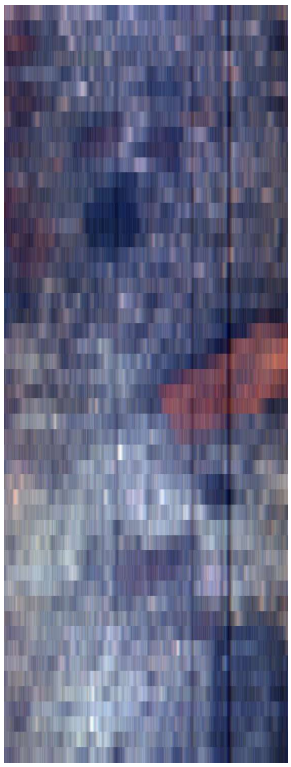
Details about the spatial registration of the two images are outside of the scope of this paper. In broad strokes, a first coarse registration was possible due to timestamp synchronization between the two cameras, then fine-tuned through an image registration method – available in MATLAB as `imregister()`.

3.2. Performance metrics and benchmark dataset

As described in [3], there is a number of metrics commonly used to evaluate the performance of enhancement methods. Those metrics are: Cross Correlation (CC), measuring the spatial enhancement, with 1 as optimal value; Spectral Angle Mapper (SAM), as the name suggests indicates spectral fidelity, 0 is ideal; Root Mean Square Error (RMSE) and Erreur Relative Globale Adimensionnelle de Synthèse (ERGAS), both global quality indices, with 0 as ideal value. In Table 1 we compare how the values change for different choices of additional terms in $\boldsymbol{C}$ – both $\boldsymbol{C}_L$ and $\boldsymbol{C}_H$.

Table 2 shows the performance of MVIF in a benchmark dataset – Moffet field, also studied in [3] – compared to the best performing method mentioned in that publication. According to its authors, that performance is achieved in a machine with an Intel Core i5 3230M 2.6GHz with 8 GB RAM. Our method was running on an Intel Core i7 4510U 2.0GHz with 8 GB RAM, so an equivalent performance is expected. Even though the results seem better – slightly better values for the quality indices, and an extreme time reduction – we do have to say that our method uses an RGB reference, while the methods discussed in that publication are using a univariate – panchromatic – high resolution reference.

(a) High resolution RGB – $\boldsymbol{C}_H$.



(b) RGB representation of low resolution HSI – $\boldsymbol{Y}_L$ – stretched to same ratio of $\boldsymbol{C}_H$.

Figure 5: RGB image from ZenMuse camera (1500 × 560 px) and RGB rendering of HS data (50 × 560 px) from Hopavågen tests.

Table 1: Performance indices of MVIF with HICO dataset. Cross Correlation (CC), 1 is ideal. Root Mean Square Error (RMSE), Erreur Relative Globale Adimensionnelle de Synthèse (ERGAS) and Spectral Angle Mapper (SAM), 0 is ideal.

Terms in $\boldsymbol{C}$	CC	RMSE	ERGAS	SAM (deg)	Time (s)
RGB	0.823	0.081	10.339	6.888	1.751
RGB, Square	0.964	0.050	6.337	5.009	1.767
RGB, Square root	0.971	0.045	4.878	5.852	1.854
RGB, Interaction	0.970	0.043	5.485	4.686	1.849
RGB, Square, Square root	0.978	0.040	4.413	5.109	1.809
RGB, Interaction, Square, Square root	0.981	0.036	4.679	3.868	1.895

3.3. Plots

Results shown here were obtained taking into consideration the values in Table 1, meaning we opted for adding Interaction, Square and Square Root terms to $\boldsymbol{C}$ before sub-

Table 2: Performance comparison in Moffett field dataset. Cross Correlation (CC), 1 is ideal. Root Mean Square Error (RMSE), Erreur Relative Globale Adimensionnelle de Synthèse (ERGAS) and Spectral Angle Mapper (SAM), 0 is ideal. Values for Bayesian Sparse method extracted from [3]. To note that Bayesian Sparse enhanced HS data using only panchromatic high resolution data, while we used RGB.

Method	CC	RMSE	ERGAS	SAM (deg)	Time (s)
Bayesian Sparse	0.982	200.158	3.426	6.625	133.61
MVIF	0.985	164.861	3.424	5.427	0.95

mitting it to the Regression step, as it gave the overall best performance. In addition, we used Non-Negative Matrix Factorization (NNMF) to unmix the hyperspectral data cubes – $\mathbf{Y}_L$ and $\widehat{\mathbf{Y}}_H$ – and make it possible to represent in low dimension. Again, we estimated the number of factors using the method in Appendix A.3. The plots with the spectral signatures for each component are matched in color with the most similar between low – Figures 6 and 8 – and high – Figures 7 and 9 – resolution for each dataset.

Notice that each of the low resolution abundance maps are enhanced to high resolution, leaving no trace of low spatial resolution artifacts.

In Figures 10 and 11 we show a factorization of the residuals that could not be enhanced. Here we opted for using Independent Component Analysis (ICA), implemented as FastICA [56], instead of NNMF since the residuals are not non-negative. ICA gives a more interpretable factorization than SVD/PCA, while dealing well with possible, or in this case likely, negative concentrations. Another reason for using this method is that it is fast, and can give us a clue regarding whether there is relevant data that we overlooked, or not. If there is some indication that we should further analyse the residuals, other methods such as MCR can also be applied. We stress the importance of doing such complementary analysis on the residuals. This creates awareness regarding the limitations of the method, and even if low resolution, these are still relevant data.

4. Discussion

Correlation in noise for artificial datasets. When RGB and low resolution HS data are artificially generated for benchmarking, they both originate from the same high resolution HS reference. Naturally, there is a concern that noise in both is correlated. We have tried to

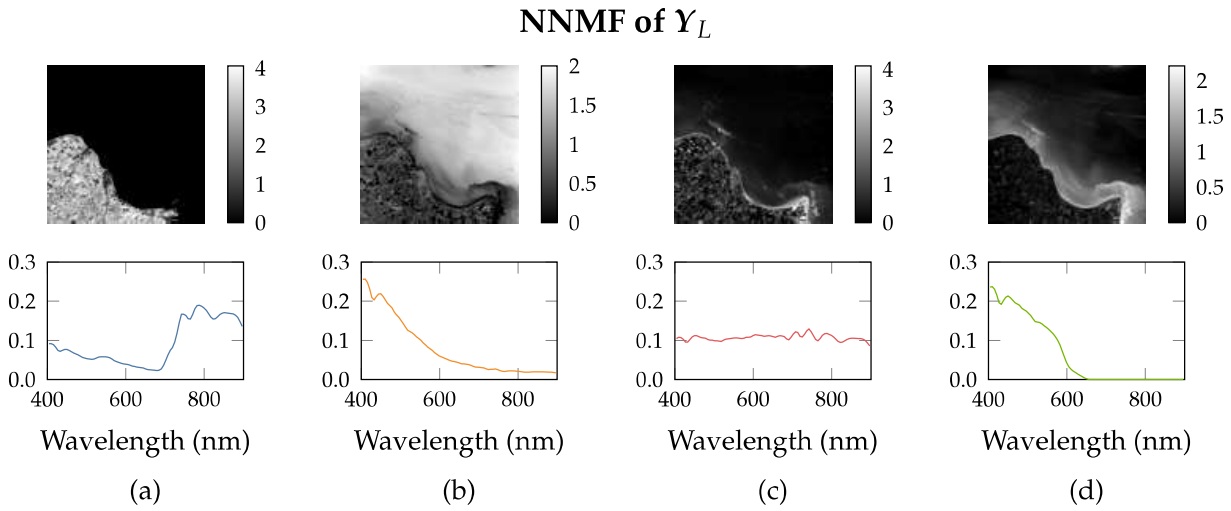


Figure 6: Non-negative Matrix Factorization (NNMF) of low resolution $\mathbf{Y}_L$. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map. The components show here can be interpreted as land based vegetation (a), some combination of CDOM (color dissolved organic matter) and phytoplankton (b and d), and an albedo-like property of pixel (c) - almost flat spectrum.

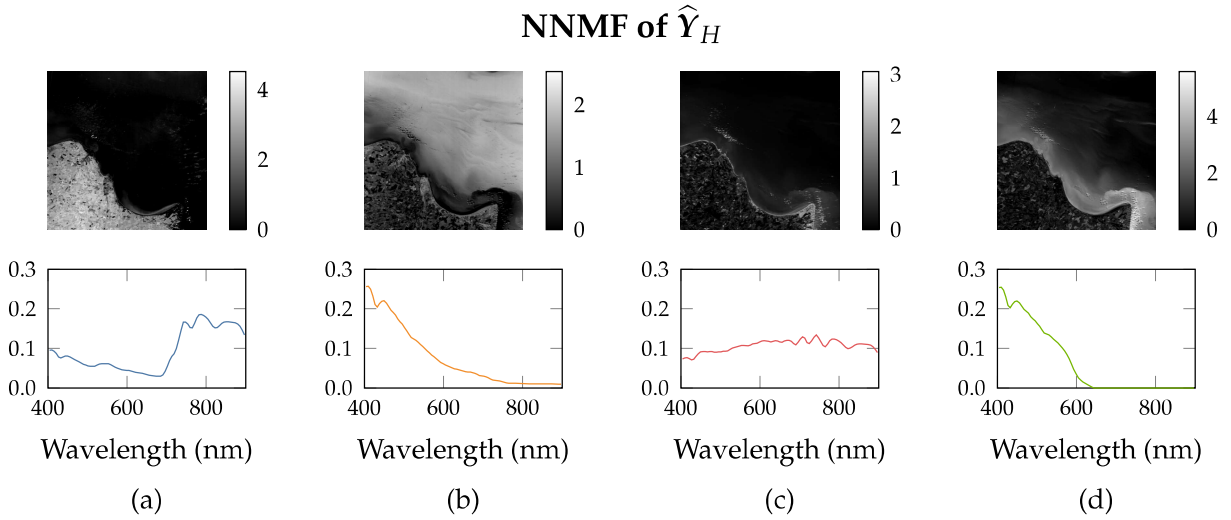


Figure 7: Non-negative Matrix Factorization (NNMF) of high resolution $\widehat{\mathbf{Y}}_H$. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map. The components show here can be interpreted as land based vegetation (a), some combination of CDOM (color dissolved organic matter) and phytoplankton (b and d), and an albedo-like property of pixel (c) - almost flat spectrum.

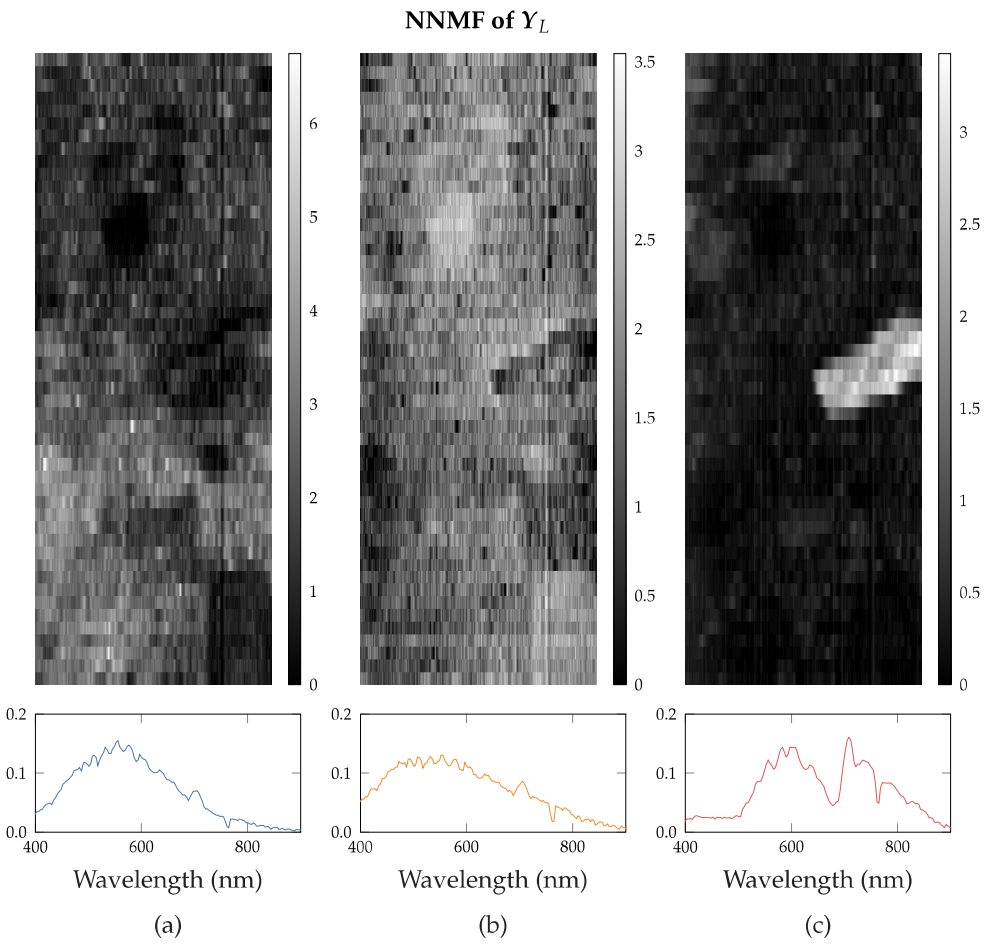


Figure 8: Non-negative Matrix Factorization (NNMF) of low resolution $\mathbf{Y}_L$. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map. In this dataset, raw data is not corrected for solar radiance. Component (a) is strongly influenced by the solar spectrum, showing what looks like intensity of reflected solar spectrum, affected both by the albedo of different materials and in-water path length – i.e. depth , which increases from bottom to top of image, due to ground slope – while (b) seems to pick out the darker rocks. Component (c) is isolating the orange/yellow rock, but the spectrum indicates some influence of chlorophyll, most likely due to the algae covering that rock. Looking at Figure 5 might help understand these components.

minimize this issue by following the convolution, blurring and downsample procedure used in [3], available at [57].

This is not an issue for our field trial dataset, as data comes from two distinct instruments.

Image registration. The work here discussed focuses on the fusion of data with different resolutions, originating from separate sensors. Here we ignore the registration problem,

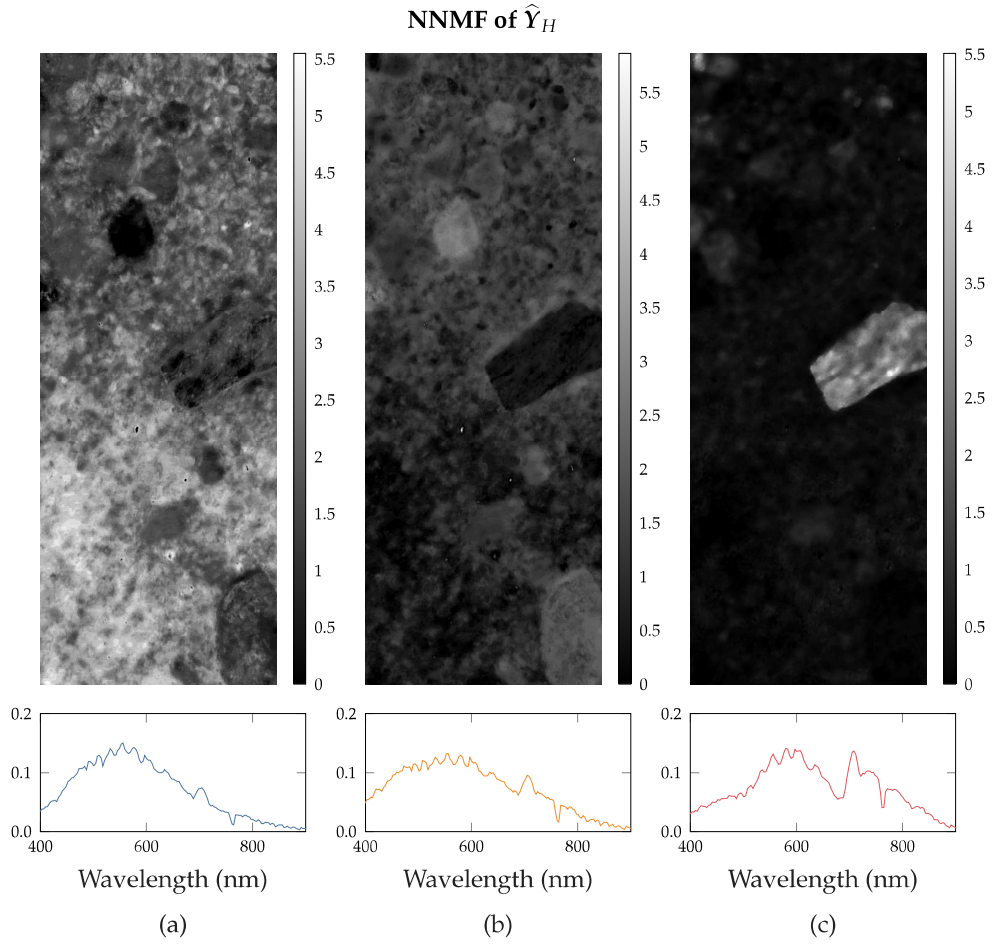


Figure 9: Non-negative Matrix Factorization (NNMF) of high resolution $\widehat{Y}_H$. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map. Notice that all the low resolution artifacts and bad pixels disappear. In this dataset, raw data is not corrected for solar radiance. Component (a) is strongly influenced by the solar spectrum, showing what looks like intensity of reflected solar spectrum, affected both by the albedo of different materials and in-water path length – i.e. depth , which increases from bottom to top of image, due to ground slope – while (b) seems to pick out the darker rocks. Component (c) is isolating the orange/yellow rock, but the spectrum indicates some influence of chlorophyll, most likely due to the algae covering that rock. Looking at Figure 5 might help understand these components.

assuming data is previously aligned and matched. A future development direction would be to integrate data matching and registration as a preprocessing step.

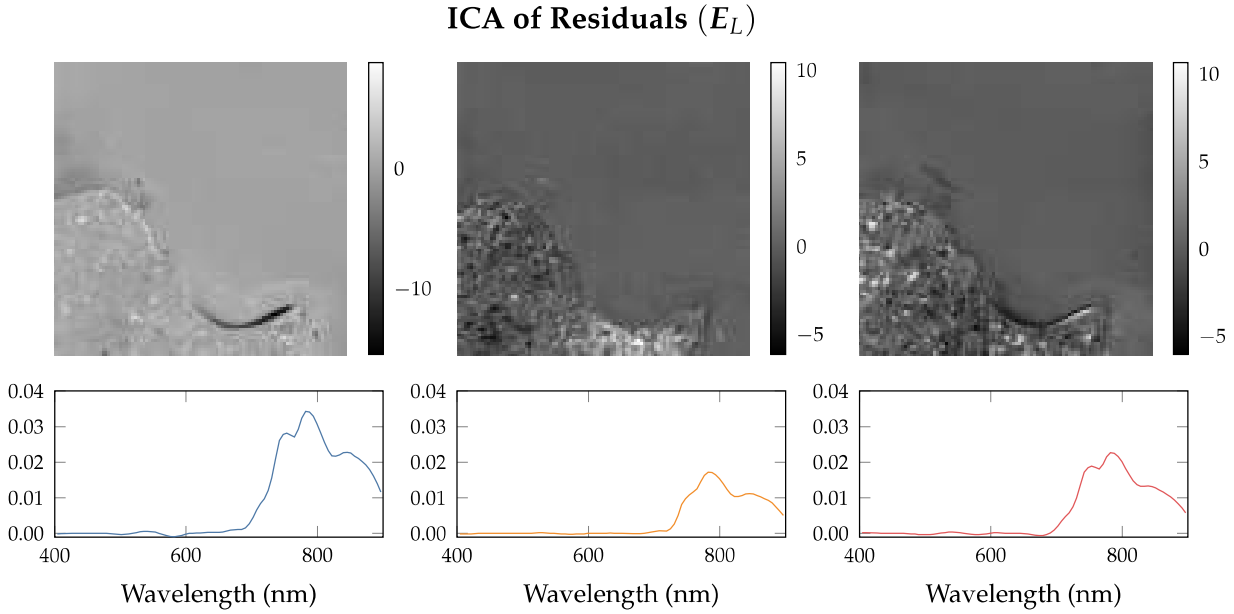


Figure 10: Independent Component Analysis (ICA) of low resolution residuals E_L. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map.

5. Conclusions

In this article we describe a pipeline for enhancing spacial resolution of HS data, taking advantage of co-located RGB data. The method is simple and fast, while giving good quality results.

Furthermore, as the epithet pipeline indicates, the method is composed by a sequence of steps. Here we describe a possible pipeline, where we use a simple projection in the Regression/Estimation step, and expand the number of variables of the RGB data in a certain way. However, the reader may find that for their application, a non-linear regression method and different variations of the high resolution data could work better.

The main contribution of this research is to provide a template for connecting functional blocks, with the aim of fusing multivariate datasets with different resolutions.

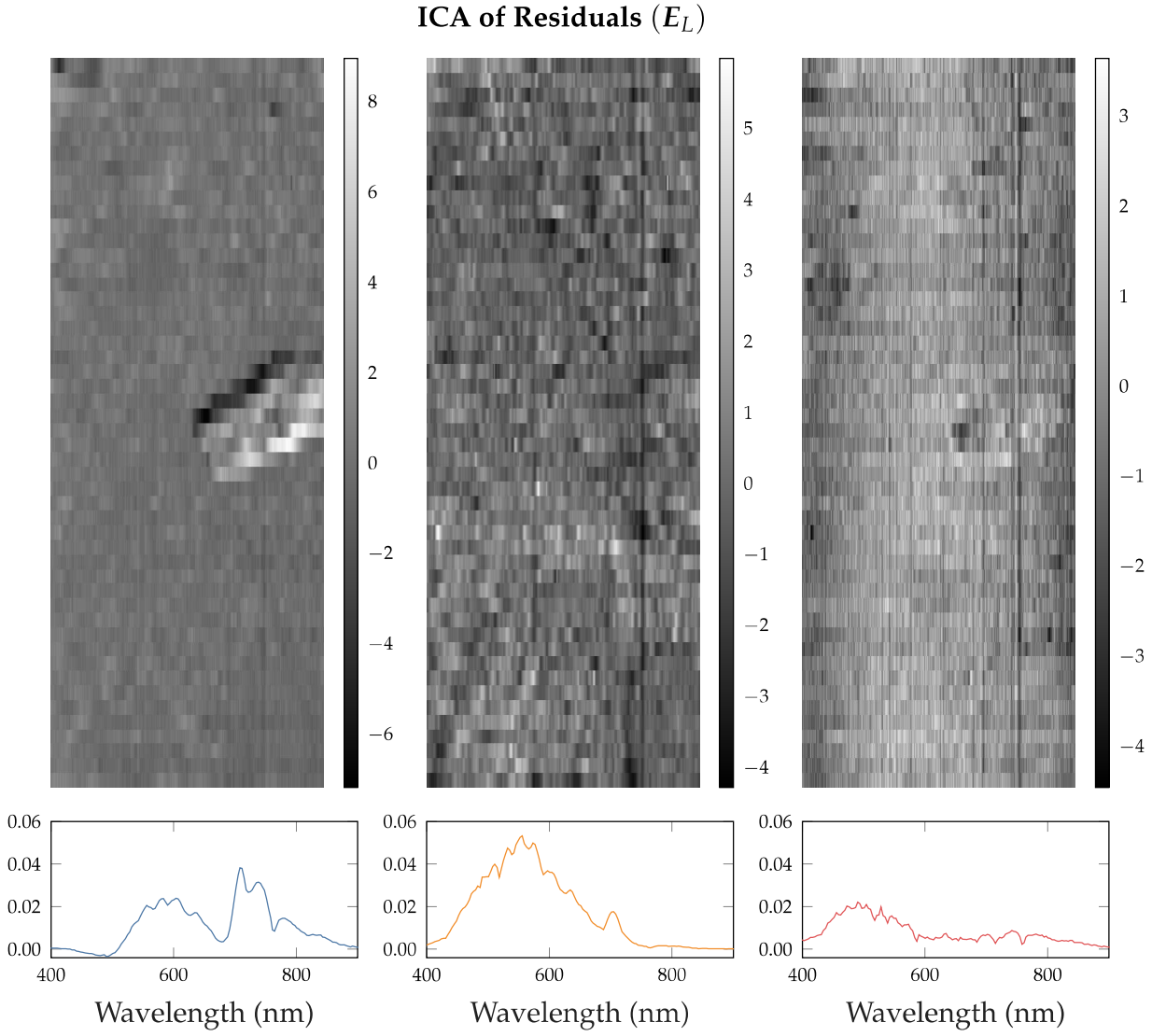


Figure 11: Independent Component Analysis (ICA) of low resolution residuals E_L. Spatial coefficients in the top row were refolded to a 2D map. The corresponding Spectral feature – systematic radiance pattern – is shown below each map.

Appendix A. Companion methods

Appendix A.1. Noise Estimation

The method for estimating the noise level per channel is very simple and intuitive, nonetheless the results match the expected instrument performance.

In essence, we check how rough, i.e. non-smooth, each band image is. Gaussian noise will cause sharp peaks very visible when taking the second difference along the horizontal and vertical axes. By looking at the absolute values, we get a direct indication of how noisy a

pixel is.

```
# Diff twice along each direction (vertical and horizontal), for each band image
diff0 = abs(diff(Y_3d, n=2, axis=0))
diff1 = abs(diff(Y_3d, n=2, axis=1))

# Flatten diff results
diff0_flat = reshape(diff0, [diff0.shape[0] * diff0.shape[1], diff0.shape[2]])
diff1_flat = reshape(diff1, [diff1.shape[0] * diff1.shape[1], diff1.shape[2]])
```

To get an overall value per band we can either use the median of all values, horizontal and vertical together:

```
# Vertical and Horizontal all together
diff = concatenate((diff0_flat, diff1_flat), axis=0)
noise = median(diff, axis=0)
```

Or average the vertical and horizontal noises:

```
# Average of Vertical and Horizontal noise levels
median0 = median(diff0_flat, axis=0)
median1 = median(diff1_flat, axis=0)
noise = (median0 + median1) / 2
```

Once the noise level per band is known, see Figure A.12, the inverse of that is used as weights.

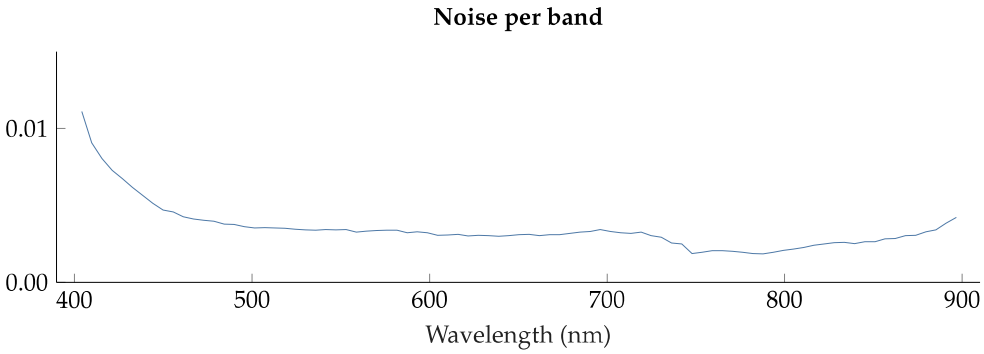


Figure A.12: Noise level per band of low resolution HICO data.

Appendix A.2. Artificial data generation

Here we describe our process to generate RGB and low resolution data, from a high resolution HS reference.

Appendix A.2.1. RGB data

Unlike common practice, instead of picking 3 bands out of the HS data cube we simulate the sensitivity of an RGB sensor. RGB sensors do not have very narrow band-pass filters for Red-Green-Blue wavelengths, so picking one single band to represent each channel would not give us a realistic dataset. Therefore, a weighted sum of different bands extracted from the HS data, with the weights based on the specification sheet of a CMOS RGB sensor, was used for each channel instead, see Figure A.13. This way we ensure,

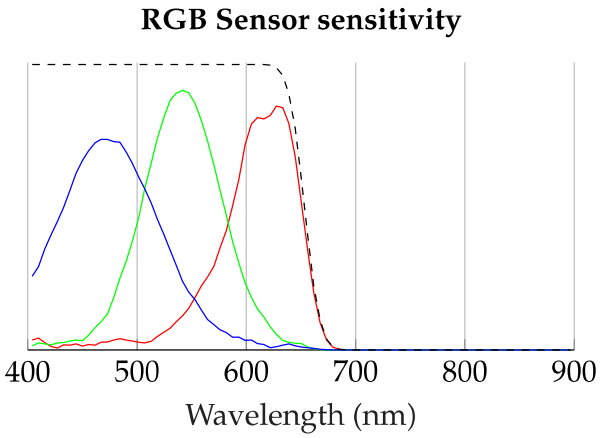


Figure A.13: Weights per band. Weights are designed to simulate an RGB sensor. The black dashed line represents the NIR cut-off filter, usually present in RGB sensors. The vertical scale is unimportant here, we merely want to show the shape of the curves.

through sensible assumptions, that the RGB data are realistic.

Appendix A.2.2. Low resolution HS data

For validation, we generated low resolution data using the same method as the study in [3], available as MATLAB code in [57].

Appendix A.3. Relevant Components

When analysing the residuals before unmixing, the first step is to decompose them through SVD. Then $\boldsymbol{U}$ (Scores), $\boldsymbol{\Sigma}$ (singular values), and $\boldsymbol{V}$ (Loadings) are evaluated by

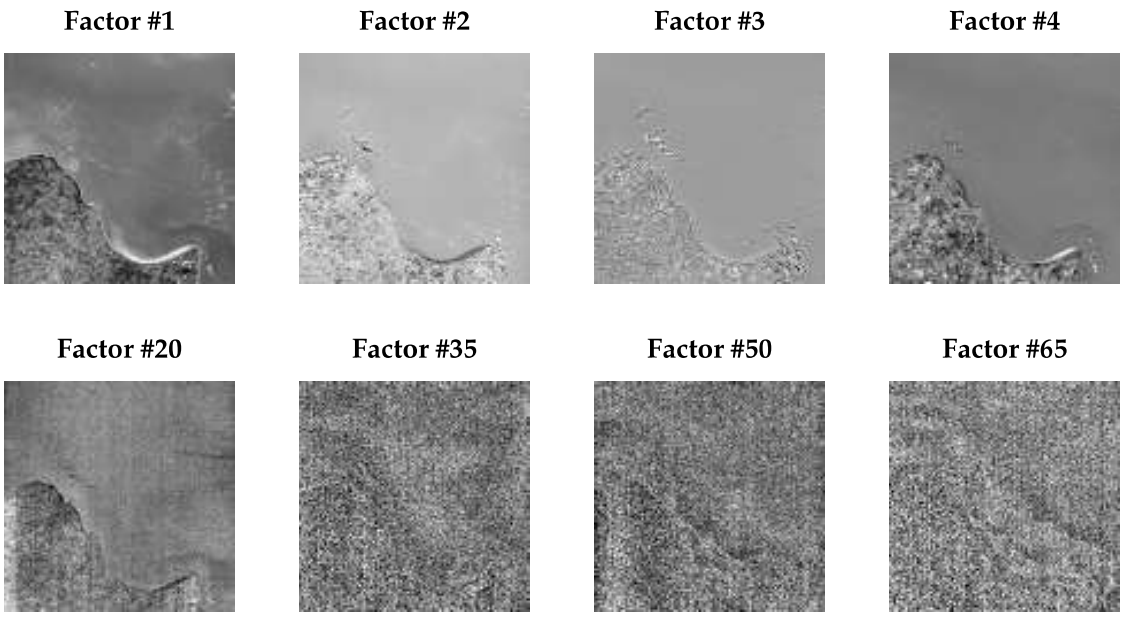


Figure A.14: Spatial Maps (Scores) of a sample of factors from the HICO dataset. Higher factors are clearly more noisy than the lower/earlier.

different methods:

1. the noise level of the spatial distribution maps ($\boldsymbol{U}$)
2. the slope of the scree plot [58] ($\boldsymbol{\Sigma}$)
3. the smoothness of the spectral signatures ($\boldsymbol{V}$)

Noise level of the spatial distribution maps. The same noise per channel routine that was used to find the noise in the raw data is re-used, now on the refolded distribution maps ($\boldsymbol{U}$). Relevant components are expected to have little noise in the spatial domain. Note that $\boldsymbol{U}$ is not scaled with $\boldsymbol{\Sigma}$, so every band has a similar range, and the same threshold can be applied. See Figure A.14

Slope of the scree plot. This is an implementation of an autonomous scree test [58], which is usually a visual inspection test. The scree plot will have an "elbow", which represents the boundary between relevant and non-relevant factors. Through a linear fit, we find the slope of the plateau that corresponds to the noise components, then we start checking lower numbered components until the slope of the linear fit starts to change, indicating we have reached the elbow. See Figure A.15.

```
# Maximum value is always 1,
# this way we can use same threshold values for different data
s /= s.max()
nc = len(s)

# More than half the components are usually noise in remote sensing data,
# so we start from the middle
curr_idx = nc // 2
z = polyfit(x=range(curr_idx, nc), y=s[curr_idx:], deg=1)

while curr_idx > 1:
    curr_idx -= 1
    z_new = polyfit(x=range(curr_idx, nc), y=s[curr_idx:], deg=1)
    fit_chg = abs(z - z_new)
    # If slope changes too much from previous fit, we are past the elbow
    if fit_chg[0] > threshold_fit:
        break
    else:
        z = z_new

p = poly1d(z)
fit_p = p(range(nc))

serr_all = (s - fit_p)**2
serr_fit = serr_all[curr_idx:]

# Relevant components will have large error to fitted line (not in flat region)
vote_slope = serr_all > threshold_error
```

Smoothness of the spectral signatures. Even though the loadings matrix (V) resulting from an SVD of hyperspectral images cannot be directly interpreted as spectral signatures of natural phenomena, which are usually smooth, they are linear combinations of smooth signatures. For sufficiently high spectral resolution sensors, which we assume a hyperspectral camera has, this means that information is smooth in the spectral domain, and noise is not. A smooth spectrum will have small variations in slope. See Figure A.16.

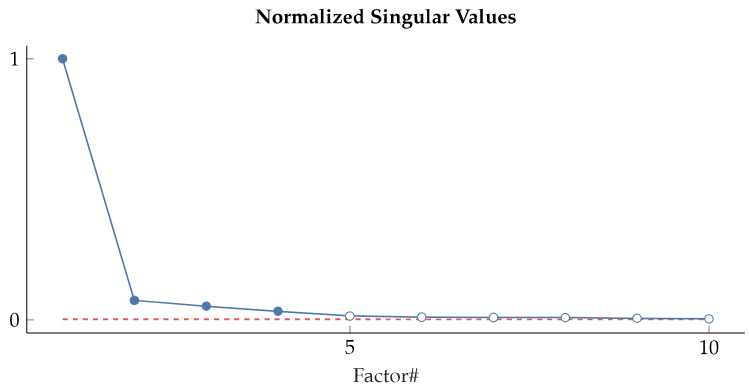


Figure A.15: Scree plot showing only the 10 first components of the low resolution residuals from the HICO dataset. Red dashed line was fitted to the flat section. When the dashed and solid lines diverge we have the relevant factors (marked with filled circles).

```
# Absolute sum of diff along wavelength axis
# large values will show for rough spectra
v_diff = diff(v, n=2, axis=1)
sum_diff = sum(abs(v_diff), axis=1)

# Relevant components have smooth spectra
vote_smooth = sum_diff < threshold_smooth
```

Once we have all the votes from the 3 classifiers, we decide how many factors to keep.

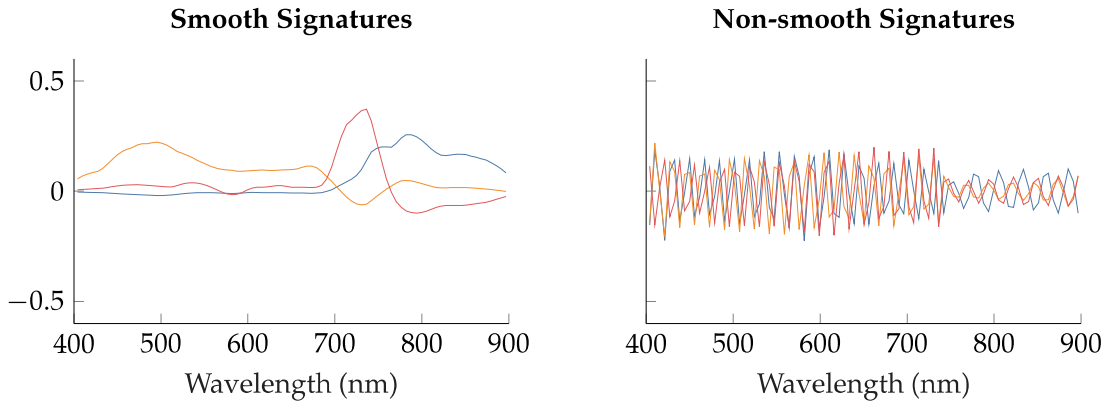


Figure A.16: Example Spectral Signatures (Loadings) split into smooth or non-smooth according to our classifier. The high frequency variations seen in the spectra on the right plot are often correlated with noise. From the HICO dataset.

Figure A.17 shows the result of such voting.

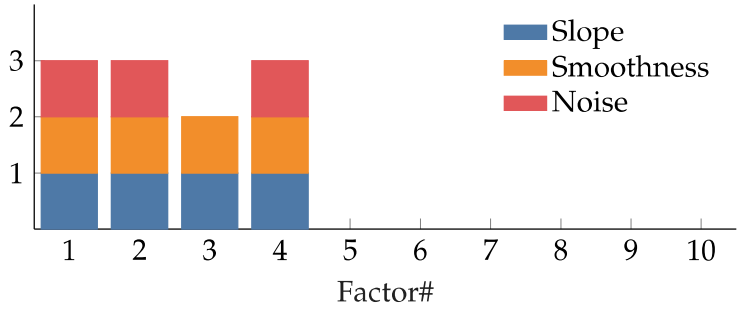


Figure A.17: Votes from each "relevance classifier" for the HICO dataset. Factors with 3 votes are considered relevant. Here, 3 relevant factors should be possible to extract from residuals.

References

[1] C. K. Singh (Ed.), Geospatial Applications for Natural Resources Management, CRC Press, 2018.

[2] F. Sigernes, M. Syrjäsuo, R. Storvold, J. Fortuna, M. E. Grøtte, T. A. Johansen, Do it yourself hyperspectral imager for handheld to airborne operations, Optics Express 26 (2018) 6021.

[3] L. Loncan, L. B. De Almeida, J. M. Bioucas-Dias, X. Briottet, J. Chanussot, N. Dobigeon, S. Fabre, W. Liao, G. A. Licciardi, M. Simoes, J. Y. Tourneret, M. A. Veganzones, G. Vivone, Q. Wei, N. Yokoya, Hyperspectral Pansharpening: A Review, IEEE Geoscience and Remote Sensing Magazine 3 (2015) 27–46.

[4] N. Yokoya, C. Grohnfeldt, J. Chanussot, Hyperspectral and multispectral data fusion: A comparative review of the recent literature, IEEE Geoscience and Remote Sensing Magazine 5 (2017) 29–56.

[5] L. Bungert, D. A. Coomes, M. J. Ehrhardt, J. Rasch, R. Reisenhofer, C.-B. Schönlieb, Blind image fusion for hyperspectral imaging with the directional total variation, Inverse Problems 34 (2018) 044003.

[6] N. Akhtar, F. Shafait, A. Mian, Bayesian sparse representation for hyperspectral image super resolution, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3631–3640.

[7] M. Simoes, J. Bioucas-Dias, L. B. Almeida, J. Chanussot, A convex formulation for hyperspectral image superresolution via subspace-based regularization, IEEE Transactions on Geoscience and Remote Sensing 53 (2015) 3373–3388.

[8] Q. Wei, J. Bioucas-Dias, N. Dobigeon, J. Tourneret, Hyperspectral and multispectral image fusion based on a sparse representation, IEEE Transactions on Geoscience and Remote Sensing 53 (2015) 3658–3668.

[9] Q. Wei, N. Dobigeon, J. Tourneret, Bayesian fusion of multi-band images, IEEE Journal of Selected Topics in Signal Processing 9 (2015) 1117–1127.

[10] Y. Zhang, S. De Backer, P. Scheunders, Noise-resistant wavelet-based bayesian fusion of multispectral and hyperspectral images, IEEE Transactions on Geoscience and Remote Sensing 47 (2009) 3834–3843.

[11] Y. Chang, L. Yan, H. Fang, S. Zhong, Z. Zhang, Weighted low-rank tensor recovery for hyperspectral image restoration, 2017.

[12] R. Dian, L. Fang, S. Li, Hyperspectral image super-resolution via non-local sparse tensor factorization, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3862–3871.

[13] K. Zhang, M. Wang, S. Yang, L. Jiao, Spatial–spectral-graph-regularized low-rank tensor decomposition for multispectral and hyperspectral image fusion, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11 (2018) 1030–1040.

[14] H. Li, L. Jing, Y. Tang, H. Ding, An Improved Pansharpening Method for Misaligned Panchromatic and Multispectral Data, Sensors 18 (2018) 557.

[15] C. I. Kanatsoulis, X. Fu, N. D. Sidiropoulos, W. Ma, Hyperspectral super-resolution: A coupled tensor factorization approach, IEEE Transactions on Signal Processing 66 (2018) 6503–6517.

[16] Y. Xu, Z. Wu, J. Chanussot, Z. Wei, Nonlocal patch tensor sparse representation for hyperspectral image super-resolution, IEEE Transactions on Image Processing 28 (2019) 3034–3047.

[17] R. Dian, S. Li, L. Fang, Learning a low tensor-train rank representation for hyperspectral image super-resolution, IEEE Transactions on Neural Networks and Learning Systems 30 (2019) 2672–2683.

[18] R. Kawakami, Y. Matsushita, J. Wright, M. Ben-Ezra, Y. W. Tai, K. Ikeuchi, High-resolution hyperspectral imaging via matrix factorization, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 2011, pp. 2329–2336.

[19] N. Yokoya, T. Yairi, A. Iwasaki, Coupled Nonnegative Matrix Factorization Unmixing for Hyperspectral and Multispectral Data Fusion, IEEE Transactions on Geoscience and Remote Sensing 50 (2012) 528–537.

[20] N. Akhtar, F. Shafait, A. Mian, Sparse spatio-spectral representation for hyperspectral image super-resolution, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, Springer International Publishing, Cham, 2014, pp. 63–78.

[21] M. Selva, B. Aiazzi, F. Butera, L. Chiarantini, S. Baronti, Hyper-sharpening: A first approach on sim-ga data, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 8 (2015) 3008–3024.

[22] C. Lanaras, E. Baltsavias, K. Schindler, Hyperspectral super-resolution by coupled spectral unmixing, in: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 3586–3594.

[23] H. Kwon, Y. Tai, Rgb-guided hyperspectral image upsampling, in: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 307–315.

[24] Q. Wei, J. Bioucas-Dias, N. Dobigeon, J. Tourneret, M. Chen, S. Godsill, Multiband

image fusion based on spectral unmixing, IEEE Transactions on Geoscience and Remote Sensing 54 (2016) 7236–7249.

[25] W. Dong, F. Fu, G. Shi, X. Cao, J. Wu, G. Li, X. Li, Hyperspectral image super-resolution via non-negative structured sparse representation, IEEE Transactions on Image Processing 25 (2016) 2337–2352.

[26] M. A. Veganzones, M. Simões, G. Licciardi, N. Yokoya, J. M. Bioucas-Dias, J. Chanussot, Hyperspectral super-resolution of locally low rank images from complementary multisource data, IEEE Transactions on Image Processing 25 (2016) 274–288.

[27] K. Zhang, M. Wang, S. Yang, Multispectral and hyperspectral image fusion based on group spectral embedding and low-rank factorization, IEEE Transactions on Geoscience and Remote Sensing 55 (2017) 1363–1371.

[28] C. Yi, Y. Zhao, J. C. Chan, Hyperspectral image super-resolution based on spatial and spectral correlation fusion, IEEE Transactions on Geoscience and Remote Sensing 56 (2018) 4165–4177.

[29] X. Han, B. Shi, Y. Zheng, Self-similarity constrained sparse representation for hyperspectral image super-resolution, IEEE Transactions on Image Processing 27 (2018) 5625–5637.

[30] L. Zhang, W. Wei, C. Bai, Y. Gao, Y. Zhang, Exploiting clustering manifold structure for hyperspectral imagery super-resolution, IEEE Transactions on Image Processing 27 (2018) 5969–5982.

[31] Y. Fu, Y. Zheng, H. Huang, I. Sato, Y. Sato, Hyperspectral image super-resolution with a mosaic rgb image, IEEE Transactions on Image Processing 27 (2018) 5539–5552.

[32] Z. Pan, H. Shen, Multispectral image super-resolution via rgb image fusion and radiometric calibration, IEEE Transactions on Image Processing 28 (2019) 1783–1797.

[33] R. Dian, S. Li, Hyperspectral image super-resolution via subspace-based low tensor multi-rank regularization, IEEE Transactions on Image Processing 28 (2019) 5135–5146.

[34] C. Wang, Y. Liu, X. Bai, W. Tang, P. Lei, J. Zhou, Deep residual convolutional neural network for hyperspectral image super-resolution, in: Y. Zhao, X. Kong, D. Taubman (Eds.), Image and Graphics, Springer International Publishing, Cham, 2017, pp. 370–380.

[35] X. Han, B. Shi, Y. Zheng, Ssf-cnn: Spatial and spectral fusion with cnn for hyperspectral image super-resolution, in: 2018 25th IEEE International Conference on Image Processing (ICIP), pp. 2506–2510.

[36] R. Dian, S. Li, A. Guo, L. Fang, Deep hyperspectral image sharpening, IEEE Transactions on Neural Networks and Learning Systems 29 (2018) 5345–5355.

[37] Y. Chang, L. Yan, H. Fang, S. Zhong, W. Liao, Hsi-denet: Hyperspectral image restoration via convolutional neural network, IEEE Transactions on Geoscience and Remote Sensing 57 (2019) 667–682.

[38] Y. Qu, H. Qi, C. Kwan, Unsupervised sparse dirichlet-net for hyperspectral image super-resolution, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2511–2520.

[39] O. Sidorov, J. Y. Hardeberg, Deep hyperspectral prior: Denoising, inpainting, super-resolution, 2019.

[40] Q. Xie, M. Zhou, Q. Zhao, D. Meng, W. Zuo, Z. Xu, Multispectral and hyperspectral image fusion by ms/hs fusion net, 2019.

[41] X.-H. Han, Y. Zheng, Y.-W. Chen, Multi-level and multi-scale spatial and spectral fusion cnn for hyperspectral image super-resolution, in: The IEEE International Conference on Computer Vision (ICCV) Workshops.

[42] Y. Zhou, A. Rangarajan, P. D. Gader, An integrated approach to registration and fusion of hyperspectral and multispectral images, IEEE Transactions on Geoscience and Remote Sensing (2019) 1–14.

[43] W. Wang, W. Zeng, Y. Huang, X. Ding, J. Paisley, Deep blind hyperspectral image fusion, in: The IEEE International Conference on Computer Vision (ICCV).

[44] NASA (@nasa) on Unsplash, Water, nature, ocean and reef, https://unsplash.com/photos/6-jTZysYY_U, 2019. Accessed: 2019-08-17.

[45] J. Fortuna, T. A. Johansen, A lightweight payload for hyperspectral remote sensing using small uavs, in: 2018 9th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), pp. 1–5.

[46] C. Bedia, À. Sierra, R. Tauler, Application of chemometric methods to the analysis of multimodal chemical images of biological tissues, Analytical and Bioanalytical Chemistry (2020).

[47] S. Piqueras, C. Bedia, C. Beleites, C. Krafft, J. Popp, M. Maeder, R. Tauler, A. de Juan, Handling different spatial resolutions in image fusion by multivariate curve resolution-alternating least squares for incomplete image multisets, Analytical Chemistry 90 (2018) 6757–6765.

[48] J. Fortuna, H. Martens, Multivariate data modelling for de-shadowing of airborne hyperspectral imaging, Journal of Spectral Imaging 6 (2017).

[49] H. Martens, T. Naes, Multivariate calibration, John Wiley & Sons, 1992.

[50] G. H. Golub, C. Reinsch, Singular value decomposition and least squares solutions, in: Linear Algebra, Springer, 1971, pp. 134–151.

[51] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (1999) 788–791.

[52] A. de Juan, R. Tauler, Multivariate Curve Resolution (MCR) from 2000: Progress in concepts and applications, Critical Reviews in Analytical Chemistry 36 (2006) 163–176.

[53] A. Hyvärinen, E. Oja, Independent component analysis: algorithms and applications, Neural networks 13 (2000) 411–430.

[54] I. T. Jolliffe, Principal Component Analysis, Springer Series in Statistics, Springer-Verlag, New York, 2 edition, 2002.

[55] Oregon State University, HICO - Hyperspectral Imager for the Coastal Ocean, http://hico.coas.oregonstate.edu/, 2019. Accessed: 2019-08-08.

[56] A. Hyvärinen, E. Oja, Independent component analysis: algorithms and applications, Neural Networks 13 (2000) 411–430.

[57] Open Remote Sensing, Hyperspectral Pansharpening : A review, https://openremotesensing.net/knowledgebase/hyperspectral-pansharpening-a-review/, 2015. Accessed: 2019-04-10.

[58] R. B. D'agostino Sr, H. K. Russell, Scree test, in: Wiley StatsRef: Statistics Reference Online, American Cancer Society, 2014.