

# POSITIVITY-BASED SEPARATION OF STELLAR SPECTRA USING A PARAMETRIC MIXING MODEL

Inès Meganem<sup>1</sup>, Shahram Hosseini<sup>1</sup>, Yannick Deville<sup>1</sup>

<sup>1</sup>IRAP, CNRS, Université de Toulouse, UPS-OMP. 14, av. Edouard Belin, 31400 Toulouse, France.  
 { ines.meganem, shahram.hosseini, yannick.deville } @irap.omp.eu

## ABSTRACT

This article presents a method to extract stellar spectra from hyperspectral images of the MUSE instrument. Because of the MUSE PSF (Point Spread Function) effect, stars are not seen as dots but spread with a certain radius. In dense fields, the information in a pixel can thus result from contributions of different stars, which requires the use of a source separation method. We first derive the mixing model using known information about the MUSE PSF, then propose a source separation method using a parametric model of the spatial PSF, the FSF (Field Spread Function). Our method is based on the positivity of the data, sources and FSF parameters. It alternately estimates the star spectra, using a least square method with positivity constraints, and the FSF parameters by a projected gradient descent algorithm. Very satisfactory results are obtained with simulated but realistic MUSE data.

**Index Terms**— Semi-blind source separation, hyperspectral images, spectra estimation, astrophysics.

## 1. INTRODUCTION

The work presented here is within the framework of the MUSE (Multi Unit Spectroscopic Explorer) project. MUSE<sup>1</sup> instrument will provide hyperspectral astrophysical images of about 3500 bands. The development of new processing methods is therefore required to permit a better exploitation of MUSE data by scientists. One of these processing methods is source separation which particularly concerns MUSE dense stellar field images. In these images, stars are not point-like objects but spread with a certain radius. This is because of the PSF (Point Spread Function) due to the instrument and atmospheric effects. As a result, each pixel in the observed image can contain contributions from several stars. We thus need a source separation method to unmix the observed spectra.

There exist three main classes of blind source separation methods [1], essentially dealing with linear instantaneous in-

variant models: methods based on ICA (Independent Component Analysis) [1], those exploiting positivity like NMF (Non-negative Matrix Factorization) [2], and those based on sparsity [1, 3]. However, as we will see in Section 2, our model is linear but spectrally variant, which makes the use of some classical methods difficult. We are also facing highly correlated sources, which excludes the use of ICA. Besides, because of the PSF spatial spreading effect, the sparsity-based source separation methods are not suitable.

We thus propose a new approach exploiting the positivity ( $\geq 0$ ) of all data (including sources, parameters) and using a parametric model for the mixing coefficients. Our method alternately performs estimation steps for the star spectra, using a least square algorithm with positivity constraints, and for the mixing parameters by a projected gradient descent algorithm.

In Section 2, we present the mixing model first introduced in [4] and show how it can be parametrized. Then, Section 3 presents our source separation method. In Section 4, we show test results with realistic simulated data.

## 2. MODELLING

### 2.1. Assumptions about the PSF

The usual shift-invariance property is not satisfied by the MUSE PSF. For an observed point-like object located at  $(z, \mu)$ , its corresponding PSF at  $(p, \lambda)$  can be denoted by  $h_{z,\mu}^{PSF}(p, \lambda)$ , where  $p$  and  $z$  are 2D spatial coordinates,  $\mu$  and  $\lambda$  are spectral coordinates. In the following, we use some relevant assumptions to derive a simplified but realistic mixing model. These assumptions, based on a recent study [5] within the framework of the MUSE project, are:

- **A1**: The PSF is separable into the spatial PSF, called FSF (Field Spread Function), and the spectral PSF, called LSF (Line Spread Function):  $h_{z,\mu}^{PSF}(p, \lambda) = h_{z,\mu}^{FSF}(p)h_{z,\mu}^{LSF}(\lambda)$ .
- **A2**: The LSF only spreads over a few spectral samples.
- **A3**: The FSF changes slowly spectrally.

### 2.2. Mixing model

We suppose stars are point-like objects in the sky before the PSF effect. Let's denote by  $e_i(\lambda)$  the spectrum of a star  $i$

<sup>1</sup>This work is part of the DAHLIA (Dedicated Algorithms for Hyperspectral Imaging in Astronomy) project, founded by the French ANR agency.

<sup>1</sup>MUSE is a new generation integral field spectrograph, working in the visible and near-infrared domain [465 nm, 930 nm]. It will be added to the VLT (Very Large Telescope), in Paranal, Chile, by the end of 2013.

located at the 2D spatial position  $z_i$ , and by  $e_i(\lambda)\delta(p - z_i)$  the contribution of this star at the 2D spatial position  $p$  (where  $\delta$  is the Dirac impulse). After the PSF effect, the contribution of this star in the observed data is:

$$\begin{aligned} y_{z_i}(p, \lambda) &= \iint e_i(\mu)\delta(z - z_i)h_{z_i,\mu}^{PSF}(p, \lambda)dzd\mu \\ &= \int_{\mathbb{R}} e_i(\mu)h_{z_i,\mu}^{PSF}(p, \lambda)d\mu. \end{aligned} \quad (1)$$

Assuming that the PSF is separable into the FSF and the LSF (Assumption **A1**), we have

$$y_{z_i}(p, \lambda) = \int_{\mathbb{R}} e_i(\mu)h_{z_i,\mu}^{FSF}(p)h_{z_i,\mu}^{LSF}(\lambda)d\mu. \quad (2)$$

Due to Assumption **A2**, the LSF can be considered equal to zero outside a small spectral interval with width  $2K$  and centered on  $\lambda$ . This yields

$$y_{z_i}(p, \lambda) = \int_{\lambda-K}^{\lambda+K} e_i(\mu)h_{z_i,\mu}^{FSF}(p)h_{z_i,\mu}^{LSF}(\lambda)d\mu. \quad (3)$$

Using Assumption **A3**, we can write  $h_{z_i,\mu}^{FSF}(p) \simeq h_{z_i,\lambda}^{FSF}(p)$  for  $\mu \in [\lambda - K, \lambda + K]$ . We thus obtain

$$y_{z_i}(p, \lambda) = h_{z_i,\lambda}^{FSF}(p) \int_{\lambda-K}^{\lambda+K} e_i(\mu)h_{z_i,\mu}^{LSF}(\lambda)d\mu. \quad (4)$$

When a pixel contains contributions from  $S$  stars, its overall value at each wavelength reads<sup>2</sup>:

$$\begin{aligned} y(p, \lambda) = \sum_{i=1}^S y_{z_i}(p, \lambda) &= \sum_{i=1}^S h_{z_i,\lambda}^{FSF}(p) \int_{\mathbb{R}} e_i(\mu)h_{z_i,\mu}^{LSF}(\lambda)d\mu \\ &= \sum_{i=1}^S m_i(p, \lambda)x_i(\lambda) \end{aligned} \quad (5)$$

where  $x_i(\lambda) = \int_{\mathbb{R}} e_i(\mu)h_{z_i,\mu}^{LSF}(\lambda)d\mu$  is the  $i^{th}$  star spectrum spectrally convolved by the LSF<sup>3</sup> and  $m_i(p, \lambda) = h_{z_i,\lambda}^{FSF}(p)$ . This yields in matrix form at each wavelength  $\lambda$  ( $\lambda$  is removed from notations for the sake of simplicity):

$$\mathbf{y} = \mathbf{M}\mathbf{x} \quad (6)$$

with

◦  $\mathbf{y} = [y(p_1, \lambda), \dots, y(p_N, \lambda)]^T$ : the vectorized observed image at wavelength  $\lambda$ , where  $N$  is the number of pixels in the considered field ( $T$  stands for Transpose).

◦  $\mathbf{x} = [x_1(\lambda), \dots, x_S(\lambda)]^T$ .

◦  $\mathbf{m}_i = [m_i(p_1, \lambda), \dots, m_i(p_N, \lambda)]^T$ , the vectorized FSF for

<sup>2</sup>Integrating the LSF over the whole spectrum or just over the interval  $[\lambda - K, \lambda + K]$  is the same because the LSF is equal to zero outside this interval.

<sup>3</sup>We call it a ‘‘convolution’’ but note that it is a misnomer because, due to the variability of the PSF, this integral does not correspond to the usual definition of a convolution.

star  $i$ .

◦  $\mathbf{M} = [\mathbf{m}_1 \dots \mathbf{m}_S]$ .

The obtained model (6) is thus linear instantaneous but spectrally variant, because the mixing matrix  $\mathbf{M}$  depends on  $\lambda$ .

In reality, the observed data are noisy. The noise can be approximated as an additive Gaussian, centred, spatially and spectrally independent process. It is non stationary and, since its generation is due to a Poissonian phenomenon, its variance is an increasing function of the signal intensity at each point: the noise level is high where the signal level is high. The entire model, for a given wavelength, can be written as follows

$$\mathbf{y} = \mathbf{M}\mathbf{x} + \mathbf{b} \quad (7)$$

with  $\mathbf{b} \sim \mathcal{N}(0, \mathbf{\Gamma})$  and  $\mathbf{\Gamma} = \text{diag}(\sigma_1^2, \dots, \sigma_N^2)$ . The noise variance depends on both pixel location and wavelength.

### 2.3. Model parametrisation

Thanks to work presented by colleagues in [5], we know that the FSF can be modelled for each source  $i$  by a Moffat function with scale and shape parameters  $\alpha$  and  $\beta$ . A coefficient  $M_{ki}$  of matrix  $\mathbf{M}$  can thus be expressed as:

$$M_{ki} = \frac{\beta - 1}{\pi\alpha^2} \left( 1 + \frac{\|p_k - z_i\|_F^2}{\alpha^2} \right)^{-\beta}, \quad \alpha \geq 0, \beta > 1 \quad (8)$$

where  $z_i$  is the spatial position of source  $i$ , and  $p_k$  the spatial position of the  $k^{th}$  pixel.

Therefore, for one given wavelength, the matrix  $\mathbf{M}$  depends on the  $(\alpha, \beta)$  parameters and on the star locations. Note that  $(\alpha, \beta)$  depend on the wavelength.

## 3. PROPOSED METHOD : LSQ-GRD

At this stage, we assume that we know the location of every star in the studied MUSE field, thanks to Hubble images<sup>4</sup>. Therefore, in the model defined by (7-8), only the spectra and the parameters  $(\alpha, \beta)$  remain unknown and have to be estimated for each wavelength. However, the final goal is only the spectra estimation.

In the following, we first describe our method in the noiseless case. We then explain in Section 3.5 how this method must be modified in the noisy case.

### 3.1. Minimised criterion

Here is the minimised criterion, defined by a Frobenius norm, for the noiseless case:

$$J = \frac{1}{2} \|\mathbf{y} - \mathbf{M}\mathbf{x}\|_F^2 = \frac{1}{2}(\mathbf{y} - \mathbf{M}\mathbf{x})^T(\mathbf{y} - \mathbf{M}\mathbf{x}). \quad (9)$$

<sup>4</sup>Hubble is a spatial telescope in orbit around the earth since 1990. It is coupled with many spectrometers. This permits it to cover a spectral domain spreading from the infra-red to near ultra-violet wavelengths.

### 3.2. Proposed algorithm for one wavelength

The proposed algorithm is iterative. The estimation of the star spectra and  $(\alpha, \beta)$  is performed alternately using a least square method with positivity constraints (for the spectra estimation) and a projected gradient descent method with a fixed step (for the  $(\alpha, \beta)$  parameters). Our global algorithm is thus called "LSQ-Grd" for Least Square - Gradient.

#### LSQ-Grd Algorithm

- Initialisation:  $(\alpha, \beta)$  (+ spectra)
- Repeat until convergence:
  - Compute matrix  $M$  using the Moffat expression with the current values of  $(\alpha, \beta)$ .
  - Estimate  $\mathbf{x}$ : by a least square method with positivity constraints, with  $M$  fixed :

$$\mathbf{x} \leftarrow \underset{\mathbf{x}}{\operatorname{arg\,min}} J, \quad \text{with } \mathbf{x} \geq 0. \quad (10)$$

- Estimate  $(\alpha, \beta)$ : by a projected gradient descent method with a fixed positive step size  $\varepsilon$ , with  $\mathbf{x}$  fixed:
  - Repeat until the gradient descent convergence:

$$\alpha \leftarrow \left[ \alpha - \varepsilon \frac{\partial J}{\partial \alpha} \right]_{P^+} \quad (11)$$

$$\beta \leftarrow \left[ \beta - \varepsilon \frac{\partial J}{\partial \beta} \right]_{P^+} \quad (12)$$

where  $[u]_{P^+}$  corresponds to the projection of  $u$  on the interval  $P^+$ .

The gradient descent here is called "projected" since, for each iteration, the values of  $(\alpha, \beta)$  are projected on the interval  $P^+ = [1, 5]$  in order to fulfil the positivity constraint and to satisfy the conditions related to our data<sup>5</sup>. In practice, if the estimated value is outside  $P^+$ , it takes the value of the nearest limit.

### 3.3. Initialisation

The above method is applicable to every wavelength separately. However, in practice, we apply it sequentially to the different wavelengths. The algorithm initialisation is thus handled as follows (for every wavelength):

- For the beginning of the global *LSQ-Grd* algorithm:
  - $(\alpha, \beta)$  are initialised, at each wavelength, with their values obtained at the previous wavelength. This permits us to accelerate convergence, since  $(\alpha, \beta)$  vary slowly spectrally. For the first wavelength, we assume knowing the interval of variations of  $(\alpha, \beta)$  and initialize randomly in this interval.
  - The explicit initialisation of spectra is not necessary, since it is handled by the Matlab '*lsqnonneg*' function used here to compute the least square estimate of  $\mathbf{x}$  in the above algorithm.

<sup>5</sup>For our studied data,  $\alpha$  and  $\beta$  vary between 1 and 2.2, we thus chose an interval  $P^+$  that satisfies this condition without being too constrained.

- For the beginning of each iteration of *LSQ-Grd*:
  - $(\alpha, \beta)$  in the gradient algorithm are initialized with their estimated values in the previous iteration of the global algorithm.
  - The spectra initialization is not necessary.

### 3.4. Gradient calculation for $(\alpha, \beta)$ estimation

We rewrite  $J$  in a scalar form for computing its gradient:

$$J = \frac{1}{2} \sum_{k=1}^N (y_k - (\mathbf{M}\mathbf{x})_k)^2 = \frac{1}{2} \sum_{k=1}^N \left( y_k - \sum_{i=1}^S M_{ki} x_i \right)^2 \quad (13)$$

with  $M_{ki}$  element  $(k, i)$  of matrix  $M$ ,  $y_k$  element  $k$  of vector  $\mathbf{y}$  and  $x_i$  element  $i$  of vector  $\mathbf{x}$ .

The derivatives of  $J$  with respect to  $\alpha$  and  $\beta$  are:

$$\frac{\partial J}{\partial \alpha} = \sum_k \sum_i \frac{\partial J}{\partial M_{ki}} \frac{\partial M_{ki}}{\partial \alpha} \quad (14)$$

$$\frac{\partial J}{\partial \beta} = \sum_k \sum_i \frac{\partial J}{\partial M_{ki}} \frac{\partial M_{ki}}{\partial \beta}. \quad (15)$$

Derivation of (13) with respect to matrix  $M$  gives:

$$\frac{\partial J}{\partial M_{ki}} = (y_k - (\mathbf{M}\mathbf{x})_k) (-x_i) = -[(\mathbf{y} - \mathbf{M}\mathbf{x})\mathbf{x}^T]_{ki}. \quad (16)$$

Some calculations then yield:

$$\frac{\partial J}{\partial \alpha} = - \left[ \frac{\partial M}{\partial \alpha} \mathbf{x} \right]^T (\mathbf{y} - \mathbf{M}\mathbf{x}) \quad (17)$$

$$\frac{\partial J}{\partial \beta} = - \left[ \frac{\partial M}{\partial \beta} \mathbf{x} \right]^T (\mathbf{y} - \mathbf{M}\mathbf{x}). \quad (18)$$

To compute  $\frac{\partial M}{\partial \alpha}$  and  $\frac{\partial M}{\partial \beta}$ , we use the expression of  $M_{ki}$  as a function of  $(\alpha, \beta)$ , given by Equation (8). We thus obtain<sup>6</sup>:

$$\begin{aligned} \frac{\partial M_{ki}}{\partial \alpha} &= \frac{2(\beta-1)}{\pi\alpha^3} \left(1 + \frac{Z}{\alpha^2}\right)^{-\beta-1} \left(\frac{\beta-1}{\alpha^2} Z - 1\right) \\ \frac{\partial M_{ki}}{\partial \beta} &= \frac{1}{\pi\alpha^2} \left(1 + \frac{Z}{\alpha^2}\right)^{-\beta} \left(1 - (\beta-1) \ln\left(1 + \frac{Z}{\alpha^2}\right)\right) \end{aligned}$$

with  $Z = \|p_k - z_i\|_F^2$ .

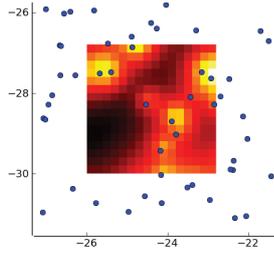
### 3.5. Noisy case

We here take into account the noise variance for each spatial location<sup>7</sup>:

$$J = \frac{1}{2} \|\mathbf{y} - \mathbf{M}\mathbf{x}\|_W^2 = \frac{1}{2} (\mathbf{y} - \mathbf{M}\mathbf{x})^T \mathbf{W} (\mathbf{y} - \mathbf{M}\mathbf{x}) \quad (19)$$

<sup>6</sup>We use:  $\frac{\partial(a^{-x})}{\partial x} = -a^{-x} \ln(a)$ .

<sup>7</sup>For noisy data, the criterion can also be written as  $J = \frac{1}{2} \|\tilde{\mathbf{y}} - \tilde{M}\mathbf{x}\|_F^2$ , with  $\tilde{\mathbf{y}} = \mathbf{W}^{\frac{1}{2}}\mathbf{y}$  and  $\tilde{M} = \mathbf{W}^{\frac{1}{2}}\mathbf{M}$ . Thus,  $\tilde{\mathbf{y}}$  and  $\tilde{M}$  can be used as the input arguments of Matlab '*lsqnonneg*' function in the noisy case.



**Fig. 1.** The studied field and its neighbourhood

with  $\mathbf{W} = (\hat{\Gamma})^{-1}$  and  $\hat{\Gamma}$  an available estimate of the diagonal covariance matrix  $\Gamma$ . We thus have:

$$\frac{\partial J}{\partial \alpha} = - \left[ \frac{\partial \mathbf{M}}{\partial \alpha} \mathbf{x} \right]^T \mathbf{W}(\mathbf{y} - \mathbf{M}\mathbf{x}) \quad (20)$$

$$\frac{\partial J}{\partial \beta} = - \left[ \frac{\partial \mathbf{M}}{\partial \beta} \mathbf{x} \right]^T \mathbf{W}(\mathbf{y} - \mathbf{M}\mathbf{x}). \quad (21)$$

## 4. TEST RESULTS

### 4.1. Data description

We work with realistic simulated data provided in the framework of the MUSE project. We here present results obtained with the small field shown in Figure 1. The coloured part of this image corresponds to the 16x16-pixel studied field and the dots show the actual star positions within the field (11 stars) and in its neighbourhood. Each pixel represents a  $0.2 \times 0.2$  arcsec<sup>2</sup> region. For this field, we have available a noiseless data cube and a noisy one characterized by a Signal to Noise Ratio (SNR) of 28 dB. An estimate of the noise variance<sup>8</sup> and the necessary data to evaluate results are available too.

Because of the FSF effect, it is important to take into account stars in the neighbourhood of the studied field to obtain a good estimation of the spectra of the stars inside the field [4]. We here take into account stars in a neighbourhood delimited by a radius of 1.4 arcsec around the field.

### 4.2. Performance criteria

To evaluate the performance, we use the following criteria:

- The total relative reconstruction error:

$$Err_{tot} = mean_{\lambda} \left( \frac{\|\mathbf{y} - \hat{\mathbf{M}}\hat{\mathbf{x}}\|_F}{\|\mathbf{y}\|_F} \right). \quad (22)$$

- The relative errors computed over the entire spectra:

$$Err_{s_i} = \frac{\|\mathbf{s}_i - \hat{\mathbf{s}}_i\|_F}{\|\mathbf{s}_i\|_F} \quad (23)$$

where  $\mathbf{s}_i$  is the entire spectrum of star  $i$ . We also use its mean:  $Err_s = mean_i(Err_{s_i})$ .

<sup>8</sup>The variance estimates will be provided by the DRS (Data Reduction Software) of MUSE.

- The relative error for the FSF:

$$Err_M = mean_{\lambda} \left( \frac{\|\mathbf{M} - \hat{\mathbf{M}}\|_F}{\|\mathbf{M}\|_F} \right). \quad (24)$$

The notations with “ $\hat{\cdot}$ ” correspond to the estimates. Errors for spectra and FSF only concern the stars inside the field.

### 4.3. Noiseless data

We first present results obtained with the noiseless data, to then better evaluate the noise impact.

Spectra are here estimated with a mean error of 0.8 %. The spectra estimation is thus almost perfect. The error over the FSF is  $Err_M = 1.4\%$ , so the  $(\alpha, \beta)$  estimation is very satisfactory. The total error is also small ( $Err_{tot} < 1\%$ ), which shows that the algorithm converges correctly.

### 4.4. Noisy realistic data

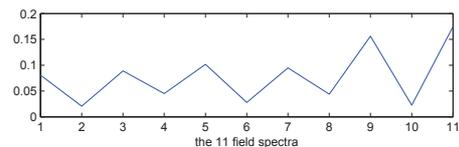
We here get for the FSF  $Err_M = 1.8\%$ , which is close to the value obtained in the noiseless case. This permits us to conclude that  $(\alpha, \beta)$  estimation is satisfactory. Concerning the spectra estimation, we can see in Figure 2 the obtained errors  $Err_{s_i}$  for the 11 stars inside the field. The mean error is 7.6%. The spectra estimation is thus less precise here and there is a disparity between the estimation qualities of the different spectra. Figure 3 shows the estimated spectra (blue) and the true ones (red)<sup>9</sup>. The spectra corresponding to a high estimation error in Fig. 2 are noisy.

We are now going to analyse the disparity in spectra estimation, which is due to noise properties.

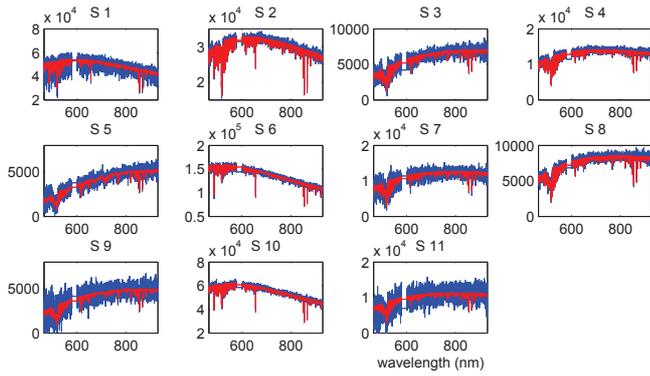
Figure 4 shows the star locations in the studied field. The star numbers here correspond to those in the previous figures. Here is our interpretation of results for the noisy estimations:

- stars 1 and 11: they are in the edge of the field, the information in the field for them is thus incomplete, which can explain the high obtained errors.
- stars 3, 5, 7 and 9: they are situated near a much brighter star (stars 2, 6, 6 and 10 respectively). The star magnitudes can be read on the axes in Figure 3. In this case, the concerned spectra are “drowned” in noise with a high level compared with their magnitude. Indeed, as stated before, MUSE noise level increases with the signal level in each pixel; at the location of a very bright star, the noise is thus high and neighbouring stars with low magnitude would be hardly well estimated.

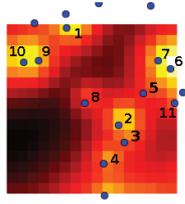
<sup>9</sup>The removed spectral bands correspond to a very noisy zone, since it has been used for the calibration with a laser star.



**Fig. 2.** Estimation error for the 11 spectra - noisy data



**Fig. 3.** Spectra inside the field: the true spectra (red) and estimated ones (blue) - noisy data

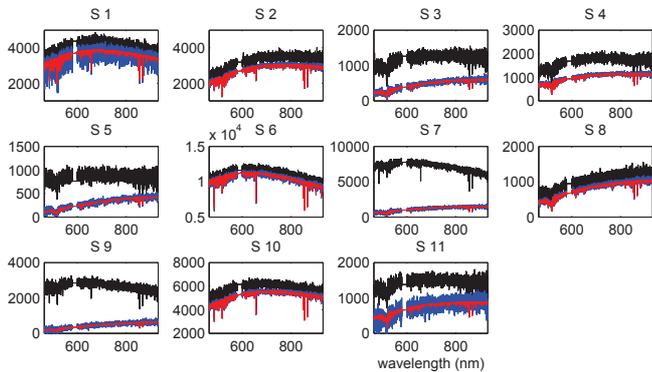


**Fig. 4.** Location of the stars in the field

Concerning the well estimated spectra, they correspond to stars that have a high magnitude or isolated stars.

To finally show the achievement of our method (even if some estimated spectra are noisy), we compare the observed (unprocessed) data and our estimates. Figure 5 shows, for each star:

- in black: the spectrum extracted from the observed data cube at the pixel corresponding to the star location.
- in red: the true spectrum, multiplied by the true FSF coefficient corresponding to the star location. This multiplication allows the magnitude of the true spectrum to be calibrated with that of the observed data (i.e. the black spectrum).
- in blue: the estimated spectrum, multiplied by the corre-



**Fig. 5.** Comparison between estimated spectra (blue), true spectra (red) and pixel spectra directly extracted from the observed cube at pixels corresponding to star location (black)

sponding estimated FSF coefficient.

Figure 5 shows that even for the most noisy estimated spectra, the estimation results are much better than the observed data. The method permits one to obtain estimated spectra that have the right magnitude and a good continuum shape. Besides, for some of the spectra, the absorption lines are well estimated. Results are thus very satisfactory for noisy realistic data.

## 5. CONCLUSION AND FUTURE WORK

In this paper, we presented a semi-blind source separation method to extract stellar spectra from MUSE dense field hyperspectral images. We started by deriving the mixing model using information about the PSF. We then proposed an adapted positivity-based method using a parametric mixing model, and validated it on realistic simulated MUSE data. As future work, we intend to add some improvements to the method, such as using a Newton version of the gradient descent algorithm. It would also be interesting to study the sensitivity of the method to errors on the star positions.

## 6. ACKNOWLEDGMENT

The authors would like to thank E. Villeneuve and H. Carfantan for their help on the PSF properties and the FSF modelling.

## 7. REFERENCES

- [1] P. Comon and C. Jutten, *Handbook of Blind Source Separation, Independent Component Analysis and Applications*, Academic Press, Oxford UK, Burlington USA, 2010.
- [2] A. Cichocki, R. Zdunek, A. H. Phan, and S.-I. Amari, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-Way Data Analysis and Blind Source Separation*, John Wiley and Sons, UK, 2009.
- [3] I. Meganem, Y. Deville, and M. Puigt, “Blind separation methods based on correlation for sparse possibly-correlated images,” in *Proceedings of IEEE International Conference ICASSP, Dallas, USA*, 2010.
- [4] I. Meganem, Y. Deville, S. Hosseini, H. Carfantan, and M.S. Karoui, “Extraction of stellar spectra from dense fields in hyperspectral MUSE data cubes using non-negative matrix factorization,” in *Proc. of IEEE international Workshop WHISPERS, Lisbon, Portugal*, 2011.
- [5] D. Serre, E. Villeneuve, H. Carfantan, L. Jolissaint, V. Mazet, S. Bourguignon, and A. Jarno, “Modeling the spatial PSF at the VLT focal plane for MUSE WFM data analysis purpose,” *SPIE Proc. Astronomical Telescopes and Instrumentation*, 2010.