Bachelor's Thesis

SPECTRAL UNMIXING OF 3D DATA FROM SELECTIVE PLANE ILUMINATION MICROSCOPY

DECONVOLUCIÓN ESPECTRAL DE DATOS 3D DE MICROSCOPÍA DE HAZ LÁSER PLANO

Author: Guillermo Lozano Sánchez

Degree: Biomedical Engineering

Professor: Jorge Ripoll Lorenzo

February 2015

Acknowledgements

First of all, I would like to thank my professor Jorge Ripoll for entrusting to me the realization of this project, and for his help and unlimited patience. Thanks Jorge for your advice and teaching, for your availability in every moment and, at last, making me see that nothing is as difficult as it seems at the beginning.

I would also like to thank my family for their unconditional support and help. Thanks to my friends and relatives who have contributed to my work and motivation in different ways.

Finally, I would like to thank my professors and university classmates, especially those who I share classes, practices and projects with. Thanks to everyone who has helped me and provided me support along my bachelor degree.

Table of Contents

List of figures

List of tables

1 Abstract

The main contribution of this bachelor thesis is the development and implementation of spectral unmixing techniques for Selective plane illumination microscopy (SPIM) images, where the usage of several fluorescent dyes with different spectrums may lead into spectral mixing, complicating the labor of distinguishing each one.

In the present work, most common techniques are reviewed and explained in detail, to select afterwards those that are more effective and simple for our purpose. First, an adequate background on spectral mixing is shown, reviewing how spectral images are obtained and how they represent information. Moreover, it is important to understand the mathematical model that represents the physical phenomenon of spectral mixing. The main approach for dealing with the mixture problem in the literature has been linear spectral unmixing, due to its advantages such as ease of implementation and flexibility in different applications. Nonlinear model was discarded in this study because of its high complexity, although nonlinear spectral unmixing may best characterize the resultant mixed spectra, especially endmember distributions where multiple scattering is present.

The main goal in this study is the spectral unmixing process, which has been faced firstly understanding the traditional methodology and its phases, being principally: dimensional reduction of the data, endmember extraction and abundance estimation. Several well-known algorithms are described and analyzed for each aim, such as Pixel Purity Index, N-FINDR or Fully Constrained Least Squares (FCLS).

On the other hand, a more simple and efficient method is proposed, based on the Linear Mixing Model and the usage of a weight matrix, representing the relationship among the fluorophores in the mixed image. Two manners were checked in order to obtain the weight matrix: one experimentally, extracting intensity values from previous images of each dye alone; and another theoretically, applying values from data sheets.

For the development of the spectral unmixing tool, Matlab has been chosen due to its intrinsic advantages like the ease to understand and learn Matlab language or its predefined functions and instructions to work with matrices. Actually, Matlab is practically standard for engineering and science studies in every university.

2 Introduction

Spectral imaging sensors often record scenes in which numerous diverse material substances contribute to the measured spectrum from a single pixel. Then, it is relevant to identify the individual constituent materials present in the mixture, as well as their proportions or abundances.

The widespread use of fluorescence microscopy along with the extensive library of fluorescent stains and staining methods has shown to be beneficial in many fields, such as material sciences or biology. The correct detection and separation of multiple stains in a sample is achieved by the use of software that executes a spectral unmixing. In addition, spectral imaging coupled with linear unmixing is becoming an increasing popular tool to solve many problems in research areas as diverse as spectral karyotyping, immunofluorescence, live-cell imaging, drug discovery and tissue pathology.

Summing up, spectral unmixing is a useful technique in fluorescence imaging for reducing the effects of native tissue auto-fluorescence and separating multiple fluorescence probes.

2.1 Motivation

Selective plane illumination microscopy (SPIM) has recently risen as one of the most popular and interesting microscopy techniques nowadays, because it is well suited for imaging deep transparent tissues or within whole organisms and it also minimizes specimen photobleaching and phototoxicity. Furthermore, the usage of several fluorescent dyes may require spectral unmixing.

The combination of spectral imaging and spectral unmixing has shown to be very useful in many microscopy modalities separating fluorescence spectral overlap in cells and tissues labeled with several fluorophores. In fact, spectral imaging coupled with unmixing

software is becoming a popular tool to solve a number of problems in research areas as diverse as spectral karyotyping, immunofluorescence, live-cell imaging, drug discovery, and tissue pathology. The spectral information provided by this technique enables to detect and differentiate between a wide variety of absorbing dyes and fluorophores. Besides, when dealing with multiple dyes in a specimen, it allows the analysis and separation of multiple absorbing dyes to generate, afterwards, single-color images that mimic how the specimen would appear with only a single stain.

2.2 Objectives

The aim of this thesis was to understand and analyze spectral unmixing method and its distinct procedures and to implement a suitable code for Selective plane illumination microscopy (SPIM).

The specific aims of this study were to:

- Conduct an extensive review of literature in the field of spectral imaging and spectral unmixing.
- Analyze and select the most appropriate algorithm for each step of the procedure.
- Implement an efficient Matlab code.
- Test and validate code, resulting software suitable for SPIM.

3 Background

In this section, the different concepts and basic terminology are described, which is important for the correct understanding of the coming lecture. The concept of multispectral image is explained, detailing its particularities and own characteristics, as well as the spectral mixing phenomenon. Next, the linear mixing model is explained in detail together with a brief annotation about the nonlinear model. Finally, Selective plane illumination microscopy (SPIM) is described.

3.1 Multispectral image

Nowadays, a complete variety of instruments and sensors are able to measure spectral singularities along different wavelengths. This fact has caused the revision of the concept of pixel. One pixel, in a gray scale image, owns a unique discrete value, while in a multispectral image it has a group of values. These values could be seen as N-dimensional vectors, being N the number of spectral bands, the sensor measures within.

From this new expanded definition, a data cube is obtained as a representation of the information, as seen in figure 1. Depending on the number of bands, a classification can be made, distinguishing multispectral images, having a low N value, and hyperspectral image with a higher magnitude of hundreds of bands.

Hyperspectral analysis focuses on the ability of sensors to provide images in a great number of nearby spectral channels, achieving a characteristic spectral signature for each material (figure 2).

The previously mentioned data cube uses two of its dimensions to represent spatial position and the third one for the spectral singularities in each wavelength.

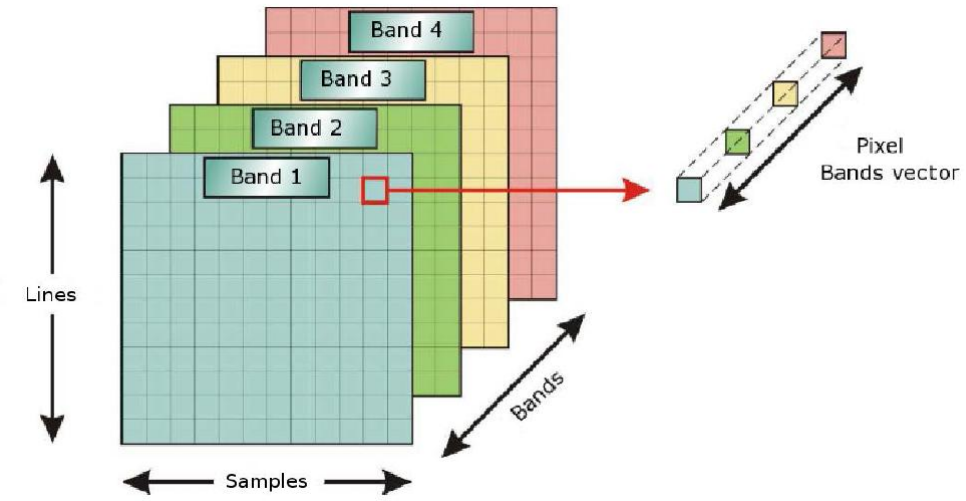


Figure 1. Multidimensional image in four bands.[1]

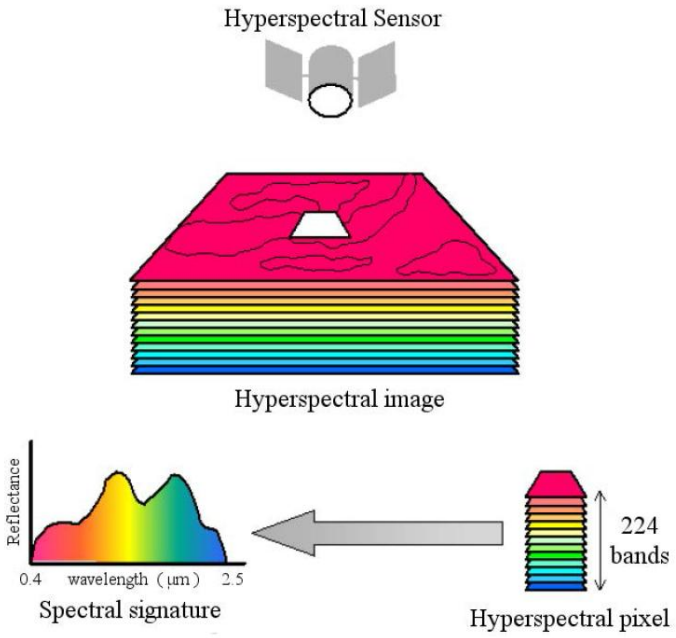


Figure 2. Hyperspectral sensor data acquisition. [1]

3.2 Spectral mixing

In multispectral imaging, mixed pixels are produced by the mixture of the spectral signature of more than one distinct material. They can be the result of the combination in homogenous mixtures of different materials. These different materials are referred as *endmembers*. Spectral unmixing is the procedure that decomposes the endmembers, or collection of constituent spectra, and the set of *abundances* or proportions of each endmember. Then, the image is represented by the endmembers and by a combination of them.

An endmember can be defined as an idealized, pure spectral signature. It should be distinguished from the concept of pure pixel, where "pixel" refers to an L-dimensional pixel vector. However, a pixel is pure if its spectral signature is an endmember.

Spectral unmixing could be understood as a generalized inverse problem, since it estimates parameters of an object observing the interaction of a signal with that object.

3.3 Linear mixing model

Mixing models are more complicated than it can seem at first sight. It describes how surface mixtures interact, but it should also take into account the effects of the three-dimensional topology of objects and the sensor observation angle. From the mixture surface, two important concepts for developing the model must be emphasized: the distinct substances, called endmembers, and the fractional abundances of each one.

The reflecting surface is represented as a checkerboard where any incident radiation interacts with one component. In the Linear Mixing Model (LMM), multiple scattering between components is not considered. This model shows the total surface area divided proportionally according to the abundances of the endmembers. Then, the reflected radiation will transmit the characteristics of the surface with the same proportions. In consequence, a linear relationship is established between the endmembers and their abundances, and the spectra of the reflected radiation.

The LMM establishes that when M endmembers exist, each one having L:

$$r = \sum_{i=1}^{M} \alpha_i m_i + w = M\alpha + w$$

where r is the $L \times 1$ received pixel spectrum vector, S is the $L \times p$ matrix whose columns are the $L \times 1$ endmembers, $m_i, \mathrm{i} = 1, \ldots, \mathrm{p}$, α is the $1 \times p$ fractional abundance vector whose entries are $\alpha_i, \mathrm{i} = 1, \ldots, \mathrm{p}$ and w is the $L \times 1$ additive observation noise vector.

The LMM is subject to two constraints on the entries of α. To be physically meaningful, all abundances must be nonnegative $\alpha_i \geq 0, \mathrm{i} = 1, \ldots, \mathrm{p}$. Second, to validate the entire composition of a mixed pixel, the full additivity condition requires $\sum_{i=1}^{p} \alpha_i = 1$.

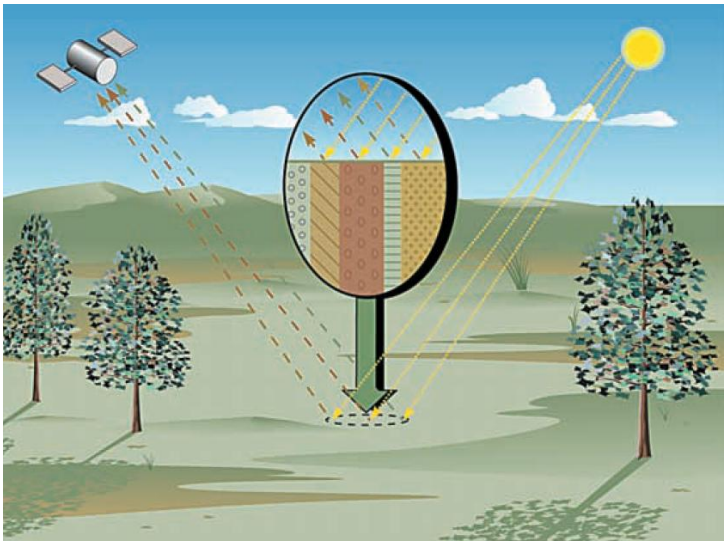


Figure 3. Illustration of linear mixing where incident solar radiation reflects from surface through a single bounce and surface consists of distinct endmembers.[2]

The nonlinear model (figure 3) has been discarded in this study because of its high complexity, although nonlinear spectral unmixing may best characterize the resultant mixed spectra for certain endmember distributions, especially where multiple scattering is present.

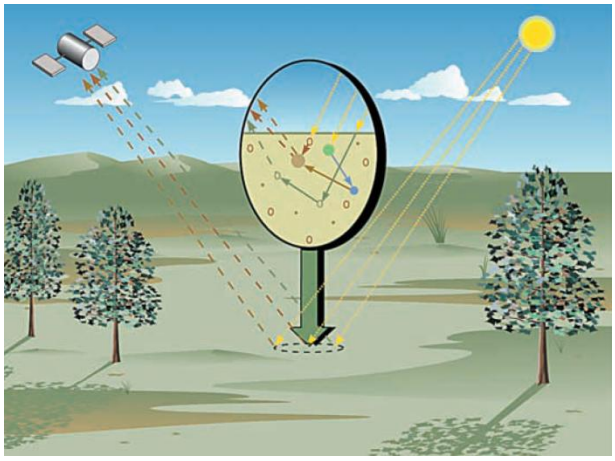


Figure 4. Illustration of nonlinear mixing where incident solar radiation encounters an intimate mixture that induces multiple bounces.[2]

3.4 SPIM

Selective plane illumination microscopy (SPIM) is a fluorescence microscopy technique that uses a focused light-sheet to illuminate the specimen from the side. Light-sheet microscopy methods are becoming increasingly popular due to their excellent resolution at high penetration depths, while being minimally invasive. Fluorescence light-sheet microscopy bridges the gap in image quality between fluorescence stereomicroscopy and high-resolution imaging of fixed tissue sections. SPIM offers several advantages over established techniques such as strongly reduced photo-bleaching, high dynamic range, and high acquisition speed.

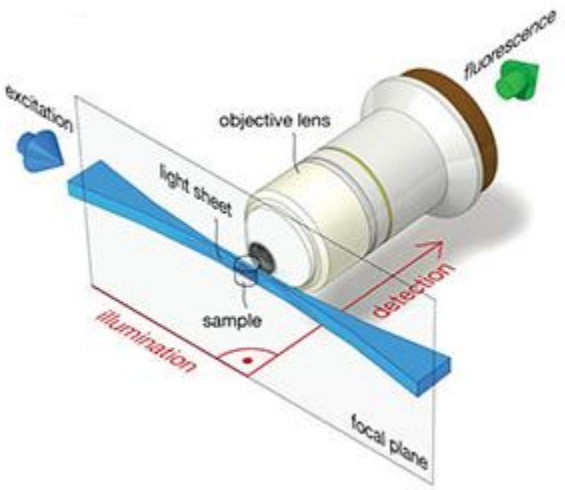


Figure 5. SPIM principle <www.mpi-cbg.de>

Focusing a thin laser light-sheet into the specimen, a perpendicular positioned detector can take 2D images of the illuminated slice. Then, a stack of 2D images is obtained just moving the specimen orthogonally to the light sheet, afterwards achieving a 3D acquisition.

4 Spectral Unmixing

In this chapter, spectral unmixing topic is covered, introducing the concept and presenting most common methods for this purpose. First, the traditional used methodology for spectral unmixing is detailed, emphasizing on the steps of the process and the most known algorithms. After this, the proposed procedure is explained for the current study on SPIM images, based on the usage of weight matrix.

4.1 Traditional Spectral Unmixing

Spectral unmixing aims to decompose a hyperspectral or multispectral scene into its constituent spectral signatures, identifying them and their fractional abundances. Because hyperspectral scenes can present extremely large volumes of data, some unmixing algorithms first reduce the dimension of the data to minimize the corresponding computation. Nevertheless, the spectral unmixing procedure is commonly divided in three stages: dimension reduction, endmember extraction and abundance estimation.

I. Dimension reduction. The dimension of the space spanned by spectra from an image is generally much lower than available number of bands. It seeks a minimal representation of the signal that sufficiently retains the requisite information for successful unmixing in the lower dimension. Afterwards, reduction improves algorithm performance and complexity, and data storage.

II. Endmember extraction. This step estimates the set of distinct spectra (endmembers) that constitute the mixed pixels in the scene.

III. Abundance estimation. Given the indentified endmembers, a constrained optimization problem is solved, estimating the fractional abundances of each mixed pixel from its spectrum and the endmember spectra.

4.1.1 Endmember extraction algorithms

Over the last decades, several algorithms have been developed for the extraction of spectral endmembers by assuming the presence of pure pixels in hyperspectral data. Wide used techniques include PPI, N-FINDER and OSP. Other advanced techniques for endmember extraction have been recently proposed considering spatial adjacency, such as AMEE.

- **PPI**

Pixel purity index is one of the most successful approaches for endmembers extraction. It is based on the geometry of convex sets and the criterion of orthogonal projection. It first generates K random unit vectors, called skewers, to cover all possible projection directors. Since its skewers are randomly produced, the results are not repeatable. Consequently, a user running PPI several times will obtain different results.

The following PPI algorithm was described by C.-I Chang et. al. (2006):

1. Find the Virtual Dimensionality (VD) to estimate the number of bands required for dimensionality reduction. (Using HFC method)
2. Apply the MNF or PCA transform to reduce dimensionality.
3. Let k be sufficiently large positive integer and use a random generator available in Matlab to produce a set of k unit vectors called skewers.

$$\{skewers\}_{j=1}^{k}$$

4. For each skewer_j all the data sample vectors are projected onto skewer_j to find sample vectors at its extreme positions to form an extrema set for the skewer$skewer_j$, denoted by $S_{extrema}(skewer_j)$. Despite the fact that a different $skewer_j$ generates a different extrema set $S_{extrema}(skewer_j)$, it is very likely that some sample vectors may appear in more than one extrema set. Define an indicator function of a set S, $I_S(r)$ by

$$I_S(r) = \begin{cases} 1; if\ r \in S \\ 0;\ if\ r \notin S \end{cases} and\ N_{PPI}(r) = \sum_j I_{S_{extrema}(skewer_j)}(r)$$

where $N_{PPI}(r)$ is defined as the PPI score of the sample vector r.

5. Find the PPI scores $N_{PPI}(r)$ for all the sample vectors defined by (1).
6. Let t be a threshold value set for the PPI score. Extract all the sample vectors with $N_{PPI}(r) \geq t$ which will be the desired set of final endmembers. Usually, the threshold t can be set to 1.

Several drawbacks can be observed from PPI algorithm. First, it is easy to see that this algorithm is not an iterative process. Therefore, it is performed for a set of k skewers and then the process is terminated. It does not guarantee that all the obtained endmembers are actually endmembers because the k skewers are randomly generated. In addition, PPI algorithm is very sensitive to noise, so NMF is crucial. Furthermore, it requires either manual selection of endmembers via visualization or the set of parameter k.

Fast Iterative PPI

In order to avoid PPI drawbacks, there are PPI-based algorithms, such as the Fast Iterative PPI (FIPPI), which iteratively searches for a better set of candidates for endmembers after each iteration. This employs several detection and classification algorithms and processes to generate an appropriate set of initial endmembers that speed up the algorithm to converge rapidly.

- **N-FINDR**

The N-FINDR method is an automated approach that identifies the endmembers as vertexes of a simplex with higher volume that can be formed from the set of points. This method is an iterative simplex volume expansion approach which assumes that, in L spectral dimensions, the L-dimensional volume formed by a simplex with vertices specified by purest pixels is always larger than that formed by any other combination of pixels. N-FINDR randomly selects a set of p pixels as initial endmembers, and calculating the volume of the simplex formed by these initial endmembers. This process is iterated through the following steps to test every pixel in the image as an endmember. The pixels that remain as endmembers at the end of the process are considered to be the final endmembers.

N-FINR employs a simplification of the data cube with as much bands as endmembers to be found. For this simplification or reduction, PCA (Principal Component Analysis) or MNF (Minimum Noise Fraction) are normally utilized.

The steps for this procedure are as follows:

1. Make a reduction of the image to get a number of bands equal to the number of endmembers, through PCA or MNF.
2. Select randomly pixels and set them as endmembers.

$$\{e_1^{(0)}, e_2^{(0)}, \dots, e_p^{(0)}\}$$

   This initial selection will be later redefined iteratively.
3. Select a pixel from the original image and exchange it successively with each of the initial endmembers.
4. Each time it is exchanged, the volume of the polygon, formed by the new point, is calculated. The volume of the simplex is calculated as an iteration k≥0 as follows:

$$V\left(e_1^{(k)}, e_2^{(k)}, \dots, e_p^{(k)}\right) = \frac{\left|det\begin{bmatrix} 1 & 1 & \dots & 1 \\ e_1^{(k)} & e_2^{(k)} & \dots & e_p^{(k)} \end{bmatrix}\right|}{(p-1)!}$$

   which is the volume of the simplex with vertices $e_1^{(k)}, e_2^{(k)}, \dots, e_p^{(k)}$, denoted by $S(e_1^{(k)}, e_2^{(k)}, \dots, e_p^{(k)})$.
5. If the new achieved volume is higher than the old one, the new pixel becomes an endmember and replaces the old one. If not, the exchange is undone.
6. Steps 4-6 are iteratively repeated until all the pixels are checked.

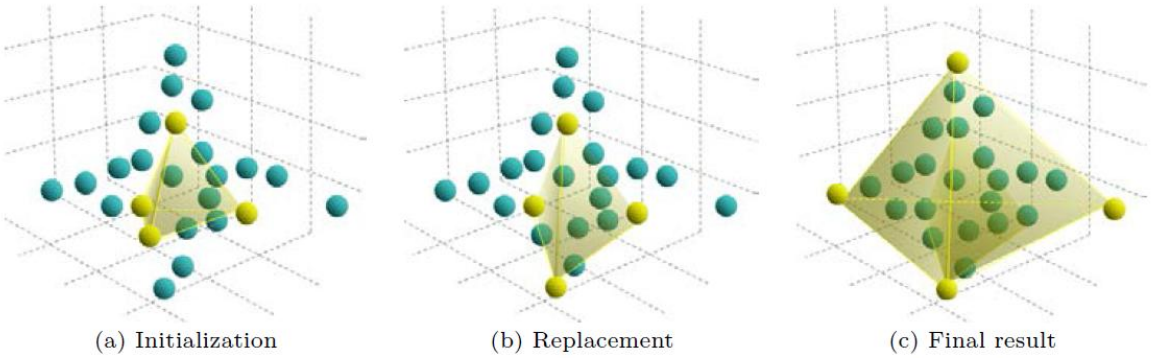


Figure 6. N-FINDR principle: The N-FINDR algorithm inflates a simplex with maximum volume using the pixels available in the input multispectral data.[1]

This procedure presents one main issue: if the initial value set for number of endmembers is too small, not all the endmembers in the data will be extracted. If that number is too large, some extracted endmembers may not be pure. This problem is solved estimating the number of endmembers previously. For this purpose, the Harsanyi-Farrand-Chang (HFC) method is employed. This method is based on a criterion called virtual dimensionality (VD), which provides a good estimate of the number of endmember signatures in a given data set. It is useful because it does not require any prior knowledge about the data.

- **SGA**

Simplex Growing Algorithm (SGA) was developed in Chang et al. (2006) and improves the commonly used N-Finder algorithm, by including a process of growing simplexes one vertex at a time until it reaches a desired number of vertices. This results in a reduction of computational complexity. Additionally, it also selects an appropriate initial vector to avoid the issue caused by random initial conditions in the N-FINDR. In fact, N-FINDR generally produces different sets of final endmembers, since different sets of randomly generated initial endmembers are used.

The SGA algorithm proceeds as follows:

1. Use the e_1 found by the first endmember selection process as the desired initial endmember pixel and set $n = 1$.
2. At $n \geq 1$ and for each sample vector r, we calculate $V(e_1, \ldots, e_n, r)$ defined by

$$V(e_1, \ldots, e_n, r) = \frac{\left| det \begin{bmatrix} 1 & 1 & \ldots & 1 & 1 \\ e_1 & e_2 & \ldots & e_n & r \end{bmatrix} \right|}{n!}$$

   which is the volume of the simplex specified by vertices $e_1, \ldots, e_n, r$ denoted by $S(e_1, \ldots, e_n, r)$. Since the matrix $\begin{bmatrix} 1 & 1 & \ldots & 1 & 1 \\ e_1 & e_2 & \ldots & e_n & r \end{bmatrix}$ is not necessarily a square matrix, a dimensional reduction technique such as PCA or MNF is required.

3. Find e_{n+1} that yields the maximum of $V(e_1, \ldots, e_n, r)$, i.e.,

$$e_{n+1} = \arg\{max_r[V(e_1, \ldots, e_n, r)]\}$$

4. Stopping rule: If $n < p$, then $n \leftarrow n + 1$ and go step 2. Otherwise, the final set of $\{e_1, e_2, \ldots, e_p\}$ is the desired p endmembers.

4.1.2 Abundance estimation

Linear spectral mixture analysis (LSMA) is a mathematical theory and widely used technique that models a data sample as linear mixtures of a finite number of basic spectral constituent with appropriate weights from which data samples can be solved by finding these weights via linear inverse problem.

This procedure is generally employed in remote sensing to estimate abundance fractions of materials present in an image pixel. It assumes there are p basic material substances, or

endmembers, $\{m_j\}_{j=1}^{p}$ that can be used to represent data sample vectors in linear forms with their corresponding abundance fractions $\{\alpha_j\}_{j=1}^{p}$ which are unknown parameters. The spectral unmixing find the best estimates of the abundance fractions. In order to produce accurate estimates, it generally requires two constraints imposed on the linear mixture model used in LSMA, which are the abundance sum-to-one constraint (ASC) $\sum_{j=1}^{p} \alpha_j = 1$, very easy to implement, and the abundance nonnegativity constraint (ANC), $\alpha_j \geq 0$ for all $1 \leq j \leq p$, which is difficult to deal with since it results in a set of inequalities and can only be solved using numerical methods. Consequently, most of LSMA-based methods are unconstrained and do not provide very accurate results.

LMSA-based methods require a priori knowledge of the spectral signatures. This knowledge is provided by endmember extraction algorithms, such as N-FINDR, previously reviewed. The selection of pure signatures is critical for successful performance of any LSMA-based method.

- **FCLS**

Fully constrained least squares (FCLS) was developed in Heinz and Chang (2001) as an abundance estimator to accurately estimate abundance fractions. FCLS is one of the best designed linear estimators in unmixing procedures. It impose both constraints: ASC and ANC.

Firstly, Nonngetivity Constrained Least Squares (NCLS) method must be described, in order to review later FCLS method, which simply adds the Abundance sum-to-one constraint (ASC).

Having in mind the linear model $r = \sum_{i=1}^{M} \alpha_i m_i + w = \boldsymbol{M\alpha} + \boldsymbol{w}$, an NCLS problem can be described by the following optimization problem:

$$Minimize\ LSE = (M\alpha - r)^T (\mathrm{M}\ \alpha - r) \quad s.t.\ \ \alpha \geq 0$$

Where LSE is used as the criterion for optimality and $\alpha \geq 0$ represents the nonnegativity constraint $\alpha_j \geq 0$ for all $1 \leq j \leq p$. Since $\alpha \geq 0$ is a set of inequalities, the Lagrange multiplier method is not suitable. Consequently, it is introduced an unknown p-dimensional positive constraint constant vector $c = (c_1, c_2, \ldots, c_p)^T$ which $c_j \geq 0$ for all $1 \leq j \leq p$ to take care of the nonnegativity constraint. Now, a Lagrangian J can be form as follows:

$$J = \frac{1}{2}(M\alpha - r)^T(M\alpha - r) + \lambda(\alpha - c)$$

With $\alpha = c$ and

$$\left.\frac{\delta J}{\delta \alpha}\right|_{\alpha_{NCLS}} = 0 \implies M^T M\alpha_{NCLS} - M^T r + \lambda = 0$$

Which results in

$$\alpha_{NCLS} = (M^T M)^{-1} M^T r - (M^T M)^{-1}\lambda$$

$$= \alpha_{LS} - (M^T M)^{-1}\lambda$$

And

$$\lambda = M^T(r - M\alpha_{NCLS})$$

These can be used to solve the optimal solutionα_{NCLS} and the Lagrange multiplier vector$= (\lambda_1, \lambda_2, \ldots, \lambda_p)^T$.

In order to solve the NCLS problem, the following iterative algorithm was proposed by Chang et. al. (2000):

1. Initialization: set the passive set $P^{(0)} = \{1,2, \ldots p\}$ and active set $R^{(0)} = \emptyset$. Set $k = 0$.
2. Compute $\alpha_{LS} = (M^T M)^{-1} M^T r$. Let $\alpha_{NCLS}^{(k)} = \alpha_{LS}$.
3. At the k-th iteration, if all components in $\alpha_{NCLS}^{(k)}$ are nonnegative, the algorithm is terminated. Otherwise, continue.

4. Let $k = k + 1$.
5. Move all indices in $P^{(k-1)}$ that correspond to negative components of $\alpha_{NCLS}^{(k-1)}$to $R^{(k-1)}$, and the resulting index sets are denoted by $P^{(k)}$ and $R^{(k)}$, respectively. Create a new index set $S^{(k)}$ and set it equal to $R^{(k)}$.
6. Let $\alpha_{R^{(k)}}$ denote the vector consisting of all components α_{LS}in $R^{(k)}$.
7. Form a steering matrix$\Psi_{\alpha}^{(k)}$ by deleting all rows and columns in the matrix $(M^T M)^{-1}$ that are specified by $P^{(k)}$.
8. Calculate$\lambda^{(k)} = (\Psi_{\alpha}^{(k)})^{-1}\alpha_{R^{(k)}}$. If all components in $\lambda^{(k)}$ are negative, go to step 13. Otherwise, continue.
9. Calculate $\lambda_{max}^{(k)} = \arg\{max_j \lambda_j^{(k)}\}$ and move the index in $R^{(k)}$ that corresponds to $\lambda_{max}^{(k)}$ to $P^{(k)}$
10. Form another matrix $\Phi_{\lambda}^{(k)}$ by deleting every column of $(M^T M)^{-1}$ specified by $P^{(k)}$.
11. Set $\alpha_{S^{(k)}} = \alpha_{LS} - \Phi_{\lambda}^{(k)}\lambda^{(k)}$.
12. If any components of $\alpha_{S^{(k)}}$ in $S^{(k)}$ are negative, then move these components from $P^{(k)}$to $R^{(k)}$.
13. Form another matrix $\Phi_{\lambda}^{(k)}$ by deleting every column of $(M^T M)^{-1}$ specified by $P^{(k)}$.
14. Set $\alpha_{NCLS}^{(k)} = \alpha_{LS} - \Phi_{\lambda}^{(k)}\lambda^{(k)}$. Go to step 3.

In summary, at the k-th iteration, NCLS algorithm begins by calculating the unconstrained solution α_{LS}. If all components are positive, the algorithm terminates. Otherwise, negative components are identified, and their indices are moved to the active set $R^{(k)}$. A duplicate of $R^{(k)}$, referred to as $S^{(k)}$, is introduced for the purpose of keeping track of the current negative components of $\alpha_{NCLS}^{(k-1)}$ during the k-th iteration. The steering matrix $\Psi_{\alpha}^{(k)}$ is formed and the Lagrange multiplier vector $\lambda^{(k)}$ is calculated, that will steer each negative component of $\alpha_{NCLS}^{(k-1)}$ back to zero. All components of $\lambda^{(k)}$ must be negative. In case there exists at least one positive component, the index that corresponds to the maximum component of $\lambda^{(k)}$ is shuffled from $R^{(k)}$to $P^{(k)}$. Since the loop from step 6 to 12 may be repeated many times during a single iteration, $S^{(k)}$ is used to check if all previously

identified indices of maximum components of $\lambda^{(k)}$ should be retained in $P^{(k)}$ or moved back to $R^{(k)}$. Once all values of $\lambda^{(k)}$ are negative, $\alpha_{NCLS}^{(k)}$ is recalculated.

In order to take care of the ASC, a new signature matrix N is introduced:

$$N = \begin{bmatrix} \delta M \\ I^T \end{bmatrix}$$

With $= (1, 1, \dots, 1)^T$ of size p and a vector s denoted by

$$s = \begin{bmatrix} \delta r \\ 1 \end{bmatrix}$$

The utilization of δ controls the impact of ASC and its value is fixed at 10^{-5}. Using these two equations, an FCLS algorithm can be derived directly from the NCLS algorithm described in the previous section by replacing M and $r$used in the NCLS algorithm with N and s.

4.2 Proposed method for Spectral Unmixing: Weight Matrix

Spectral unmixing methods exploit the fact that the detected fluorescence signal can be expressed as a linear combination of the different fluorescent components in the sample. So besides the previous complex procedure, there is a more simple and efficient one, just following the Linear Mixing Method, already implement by M. Simantiraki et. al. (2009). So, for each detection channel, a linear equation can be derived that consist of the sum of concentration of the fluorescence emitters multiplied by a weighting factor corresponding to the strength of the emission in that channel.

Hence, being the number of detection channels equal to the number of fluorescence targets, a square system of equations can be derived and solved as follows:

$$I_1(\lambda_{exc1}) = C_{f1} \cdot w_{f1}^{bw1}(\lambda_{exc1}) + C_{f2} \cdot w_{f2}^{bw1}(\lambda_{exc1})$$

$$I_2(\lambda_{exc2}) = C_{f1} \cdot w_{f1}^{bw2}(\lambda_{exc2}) + C_{f2} \cdot w_{f2}^{bw2}(\lambda_{exc2})$$

where I_1, I_2 are the fluorescence reconstructions in the detection channels for fluorescent targets $1, 2$; C_{fi} are the unknown reconstructed concentration of fluorophore i; and $w_{fluo}^{bw\,i}$ is the spectral strength of the fluorescence target on bandwidth i, with excitation wavelengths λ_{exc1}, λ_{exc2}. In matrix notation we obtain:

$$\begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} w_{f1}^{bw1}(\lambda_{exc1}) & w_{f2}^{bw1}(\lambda_{exc1}) \\ w_{f1}^{bw2}(\lambda_{exc2}) & w_{f2}^{bw2}(\lambda_{exc2}) \end{bmatrix} \times \begin{bmatrix} C_{f1} \\ C_{f2} \end{bmatrix}$$

Then, the solution of the system is:

$$\begin{bmatrix} C_{f1} \\ C_{f2} \end{bmatrix} = \begin{bmatrix} w_{f1}^{bw1}(\lambda_{exc1}) & w_{f2}^{bw1}(\lambda_{exc1}) \\ w_{f1}^{bw2}(\lambda_{exc2}) & w_{f2}^{bw2}(\lambda_{exc2}) \end{bmatrix}^{-1} \times \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}$$

or

$$[C] = [W]^{-1} \times [I]$$

This Weight Matrix W can be achieved empirically or theoretically.

- **Theoretical procedure**

Spectral strengths along bandwidth j, with excitation wavelength λ_{exc}, could be described as:

$$w_{fi}^{bwj}(\lambda_{exc}) = \int_{bwj} Exc(\lambda_{exc}) \cdot Em(\lambda) d(\lambda)$$

Where $Em(\lambda)$ is the emission spectrum of fluorosphore fi at wavelength λ. For this procedure, spectrum tables of the used fluorescent dyes are needed. They can be easily

obtained from manufacturer's website or from other sites on the internet, and shows the intensity values for each excitation and emission wavelengths (see Appendix). Being discrete emission and excitation values, the calculation of a single strength is derived as follows:

$$w_{fi}^{bwj}(\lambda_{exc}) = \frac{1}{N} Exc(\lambda_{exc}) \sum_{i=0}^{N} Emission_i(\lambda)$$

In Matlab environment, this is summarized as:

$$w(i,j) = mean(Exc * Emm(:))$$

- **Empirical procedure**

Spectral strengths $w(i,j)$ can be easily derived empirically. For this purpose, it is necessary to have previously available the spectral image of each fluorosphore at the same excitation and emission wavelengths as the mixed image. Neglecting the background and selecting the region of interest of the sample, the mean value may represent the spectral strength of the fluorosphore corresponding to those specific wavelengths.

In order to select the more reliable values, a mask might be employed, which consists of a matrix of the same size of the images and contains 1 (corresponding to the region of interest) and 0 for the rest. The forward application of the mask is shown in figure 7.

Then, in Matlab, the calculation of the spectral strength is reduced to:

$$w(i,j) = mean(Image(mask > 0))$$

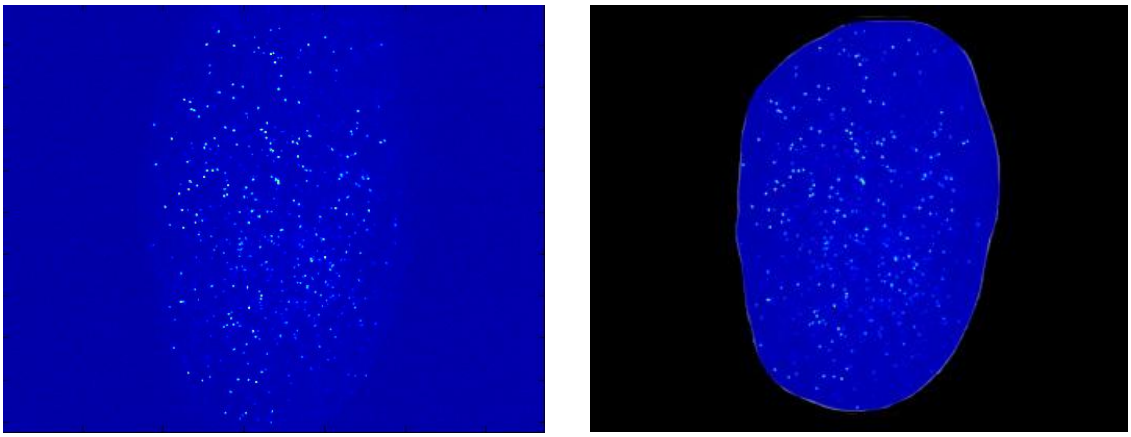


Figure 7. Experimental procedure's mask.

Since Weight Matrix W is not necessarily square, when calculating its inverse $[W]^{-1}$ it becomes appropriate to employ the Moore-Penrose pseudoinverse. This calculation may proceed using Matlab function $pinv$ or Singular Value Decomposition (SVD). Besides $pinv$ function offers an easy way to calculate the pseudoinverse, SVD allows the introduction of a regularization parameter for obtaining better results afterwards. Being the singular value decomposition of W:

$$[U, S, V] = svd(W)$$

which fulfills

$$W = USV^T$$

Then, the pseudoinverse of W, W^+, is:

$$W^+ = V^T S^{-1} U$$

and

$$S^{-1} = \left(\frac{1}{S - \lambda}\right)^T$$

where λ is the regularization parameter which allows selecting which value from the diagonal matrix S is more significant for the resulting fitting.

5 Detailed methodology

In this section, the methodology that has been followed through the realization of this study is explained in detail. Concretely, the previously proposed method is applied on Matlab programming environment, as well as both of its variations. Then, the procedure is as follows:

1. Firstly, the excitation and emission wavelengths, significant for the unmixing are selected, as well as the diameters of the employed filters.

```
Pairs_exc_emm_demm = [561, 593, 46; ...
                      561, 628, 40; ...
                      561, 670, 30; ...
                      593, 628, 40; ...
                      593, 670, 30];
```

2. The fluorophores which are present in the mixed image are introduced, so then relevant information could be gathered either data files for the theoretical method or images as experimental sources for the other procedure. Also, the files folders containing the data set are specified.

```
Fluorophores = {'Alexa594','CMTMR'};

folder = 'D:\...\20140725_SpectralUnmixing_AL594-
CMTPX-CMTMR\CMTMR+Alexa594';
```

3. The number of measurements and of fluorophores are set.

```
Nmeas = size(Pairs_exc_emm_demm,1);
Nfluo = length(Fluorophores);
```

- **Theoretical procedure**

4. From the data set, previously introduced into function Matlab files, the intensity values are extracted for each flurorophore and every excitation wavelength and emission strip, which were detailed at the beginning. Each intensity value is then introduce into the weight matrix.

```
M=zeros(Nmeas,Nfluo);
for ii=1:Nfluo
   for im=1:Nmeas
         %Select emission bandwidth and excitation wavelength
         wav_emm=(Pairs_exc_emm_demm(im,2)-
         (Pairs_exc_emm_demm(im,3)/2):
         Pairs_exc_emm_demm(im,2)+(Pairs_exc_emm_demm(im,3)/2)
         );
                     wav_exc=Pairs_exc_emm_demm(im,1);

         %Find fluorosphore spectrum data

         fin_spec=str2func(strcat('spectra_',Fluorophores{ii})
         );

         [Emm,Exc] = fin_spec (wav_emm,wav_exc);

         %Introduce Factor into Weight Matrix
         M(im,ii)=mean(Exc*Emm(:));
   end
end
```

- **Experimental procedure**

4. The mask file is loaded, in order to select only significant data.

```
Umask = imread(maskfile);
```

5. Now, every image of each fluorophore alone is read for each emission and excitation combination. Then, the mean of every image is calculated, but

only for those pixels where the mask is not zero, and this value is introduced into the weight matrix.

```
for ii=1:Nfluo
      fluo_folder = [fileparts(folder) '\'
      Fluorophores{ii}];

      for im=1:Nmeas
            exc = num2str(Pairs_exc_emm_demm(im,1));
            emm = num2str(Pairs_exc_emm_demm(im,2));

            dd = dir([fluo_folder '\*' exc '*' emm
            '*.tif']);
            filename = [fluo_folder '\' dd(1).name];

            disp([exc '; ' emm ' > ' filename]);

            Iback = imread(filename,1);
            I = imread(filename,2);
            U = double(I);% - double(Iback);
            M(im, ii) = mean(U(Umask>0));
      end
end
```

6. The pseudoinverse is calculated.

```
Minv = pinv(M);
```

7. Finally, the reading of the mixed images is performed and loaded into the matrix which will be multiplied by the inverse of the weight matrix.

```
%% READING DATA
exc = num2str(Pairs_exc_emm_demm(1,1));
emm = num2str(Pairs_exc_emm_demm(1,2));
dd = dir([folder '\*' exc '*' emm '*.tif']);
filename = [folder '\' dd(1).name];
Iback = imread(filename,1);
I = imread(filename,2);
U = double(I);
```

```
Idata = zeros(Nmeas,numel(U));

for im=1:Nmeas
      exc = num2str(Pairs_exc_emm_demm(im,1));
      emm = num2str(Pairs_exc_emm_demm(im,2));

      dd = dir([folder '\*' exc '*' emm '*.tif']);
      filename = [folder '\' dd(1).name];

      disp([exc '; ' emm ' > ' filename]);

      Iback = imread(filename,1);
      I = imread(filename,2);
      Idata(im,:) = I(:);
end
```

8. Afterwards, the resultant image is reshaped in order to have the same dimensions as the original image. Also, infinite or negative values are rejected and become zero.

```
Iunmix = Minv*Idata;
Iunmix = permute(Iunmix,[2 1]);
Iunmix = reshape(Iunmix,size(I,1),size(I,2),Nfluo);

Iunmix(~isfinite(Iunmix)) = 0;
Iunmix(Iunmix<0) = 0;
```

6 Results

In this section, here are presented the results from the application of the spectral unmixing methods described in the current study. Firstly, the images employed during the experiment are presented, as well as the details of the methodology for the further analysis. After that, an error study of the resultant unmixed images is performed, followed by a discussion about the compared methods along the present work.

6.1 Considered images for the present study

- **Real images**

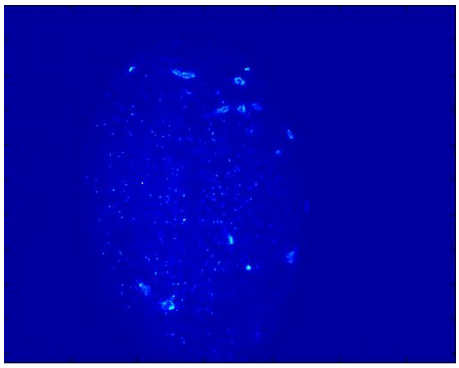
Dr. Jun Abe and Prof. Jens Stein, from Theodor Kocher Institute (University of Bern) provided SPIM images (figure 8). The used dyes were Alexa594, CellTracker Orange/CMTMR, and CellTracker Red/CMTPX.

The procedure for acquiring the SPIM images is summarized as follows. First, isolated lymphocytes were stained with CellTracker dyes and then injected into other mice. For Alexa595, a labeled antibody was injected into mice to stain specialized blood vessels called high endothelial venules (HEVs).

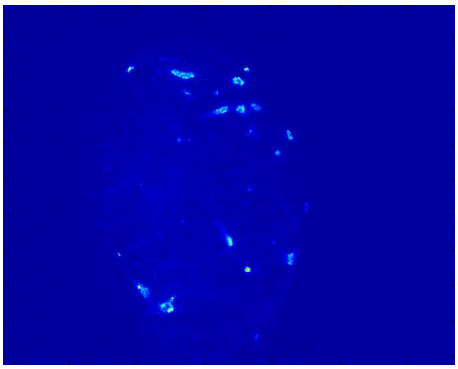
Thus, in images where appropriate excitation and emission filter sets were used, signals from CellTracker dyes appear as dots while Alexa594 signals look like vessels on a typical cryosection.

Images were taken employing several excitation laser wavelengths through different bandwidths. Images of each fluorophores were taken separately and then mixed images, using a combination of two fluorophores. Each image was taken using a combination of excitation wavelength and emission filter, which are shown in table 1.

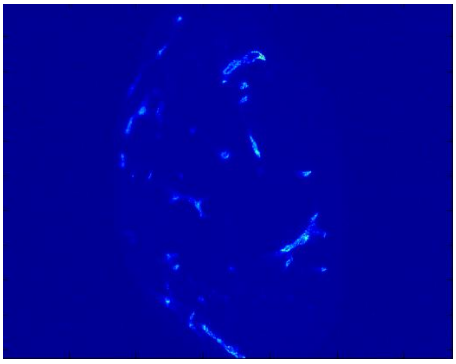
Alexa594 + CMTMR Ex561 Em593



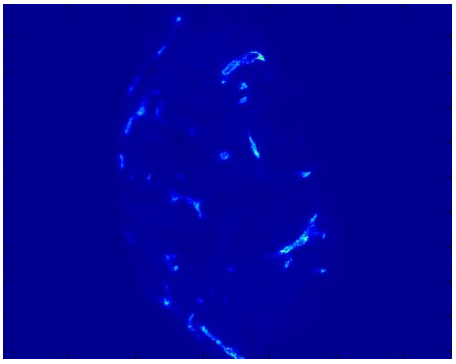
Alexa594 + CMTMR Ex561 Em628



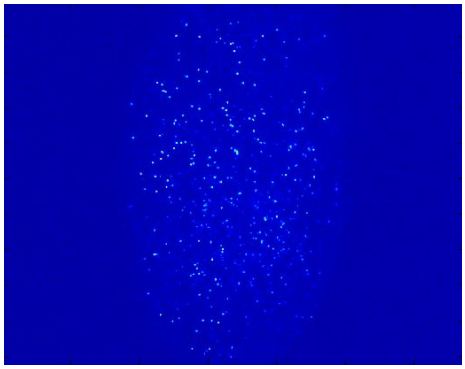
Alexa594 Ex561 Em593



Alexa594 Ex561 Em628



CMTMR Ex561 Em593



CMTPX Ex561 Em593

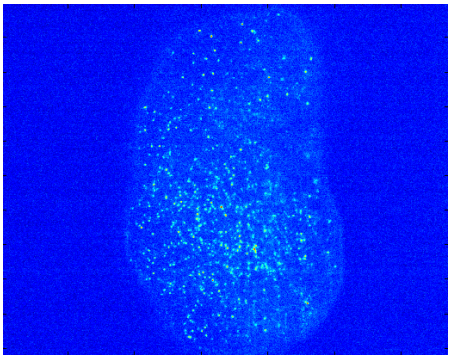


Figure 8. Sample of real images used along the experiment.

Excitation (nm)	405		561		593	635
Emission (nm)	447	490	525	593	628	670
Filter bandwidth (nm)	60	40	50	46	40	30

Table 1. Wavelengths used along the experiment.

- **Synthetic images**

In the present study, synthetic images, as well as real SPIM images, have been considered. The main advantage of using synthetic images is that the original concentration of each fluorophore is already known. Then, the employ of synthetic images allows a more reliable analysis of the effectiveness and functioning of each method. While the pattern of the synthetic images is completely artificial, the values of the pixels have been generated following the intensity data of the fluorophores.

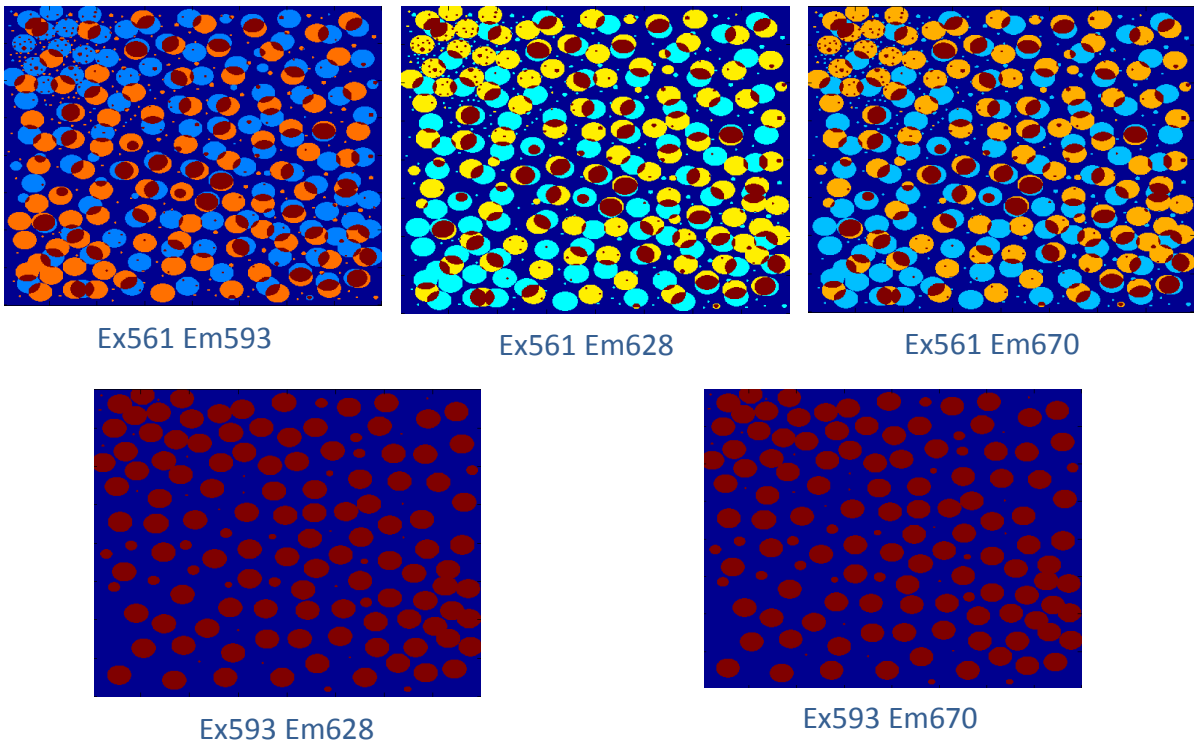


Figure 9. Sample of synthetic mixed images considered for the study.

6.2 Analysis methodology

The effectiveness and robustness of the unmixing methods have been analyzed thought the error of the resultant images compared to the original ones, which are those with one fluorophore alone. The error between the resultant unmixed images and "original" ones has been measured as follows:

$$Error = \frac{mean(|I_{original} - I_{result}|)}{mean(I_{original})}$$

On the other hand, in order to check the efficacy of the proposed method, "original" images have been simulated using the images that were taken previously for each fluorophore alone. The images considered as original are shown below:

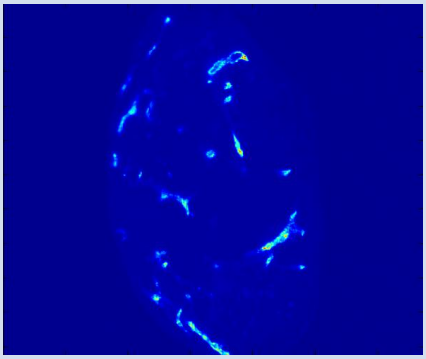
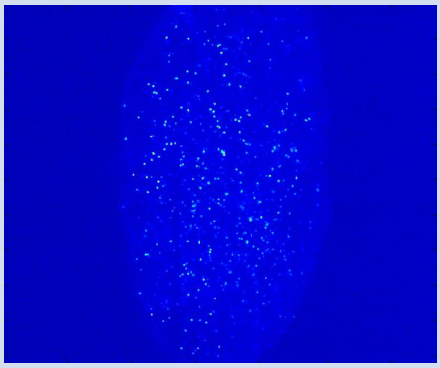
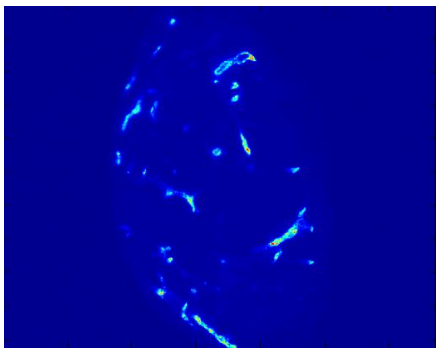
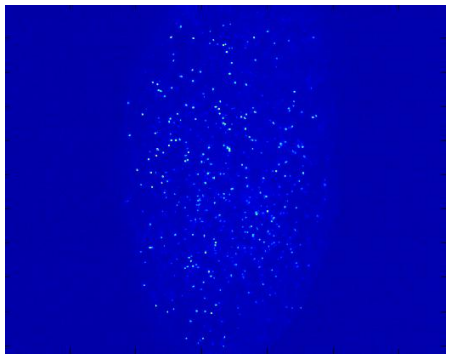
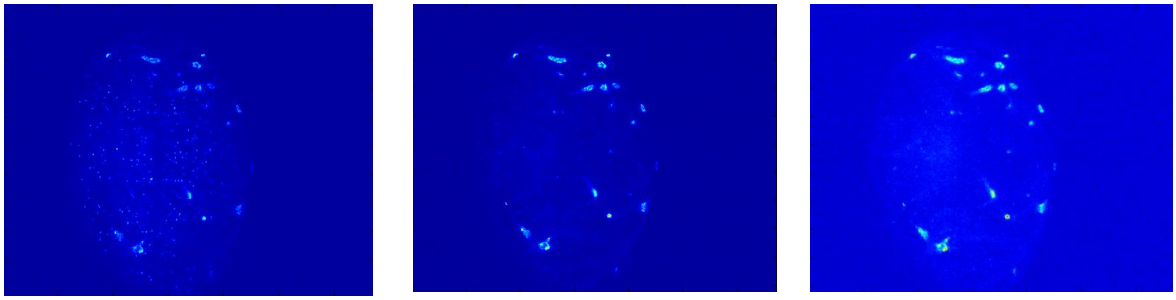
Method	Alexa594	CMTMR
Experimental		
Theoretical		

Figure 10. Images considered for the analysis of CMTMR and Alexa594 unmixing.

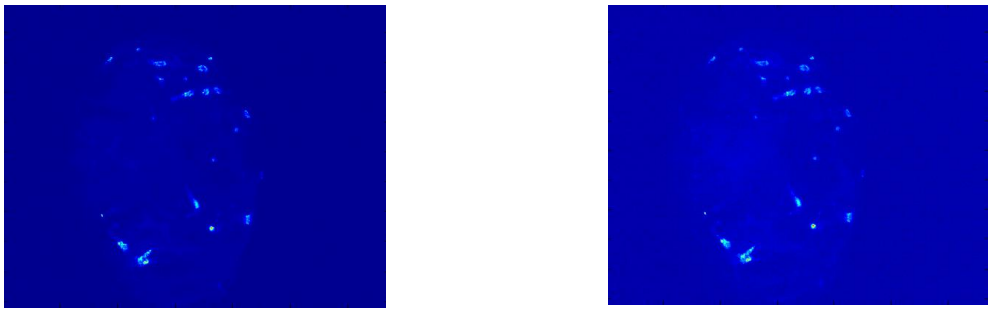
6.3 Analysis results and discussions

Firstly, wavelengths of relevance for the combination of both fluorophores were chosen and introduced into the unmixing code. Then, the real mixed images were treated and the resultant images were successfully unmixed, as shown on figures 11-13.

CMTMR + Alexa594



Ex561 Em593 | Ex561 Em628 | Ex561 Em670



Ex593 Em628 | Ex593 Em670

Figure 11. Original mixed images.

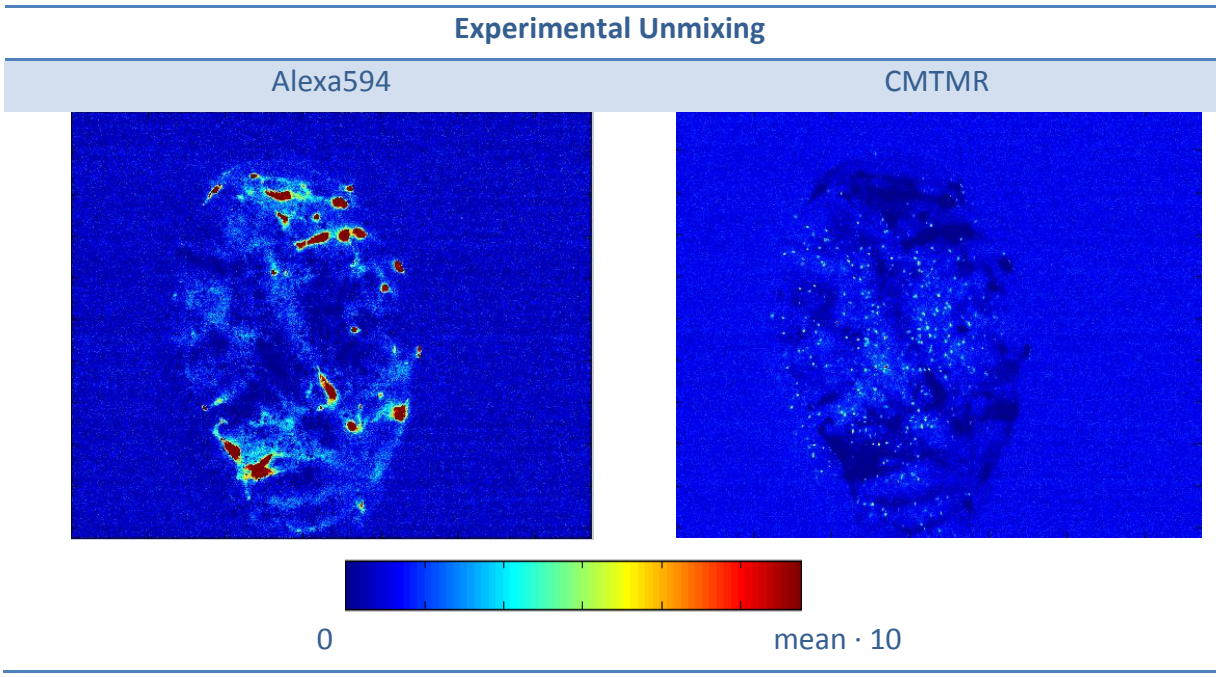


Figure 12. Unmixed images from experimental method.

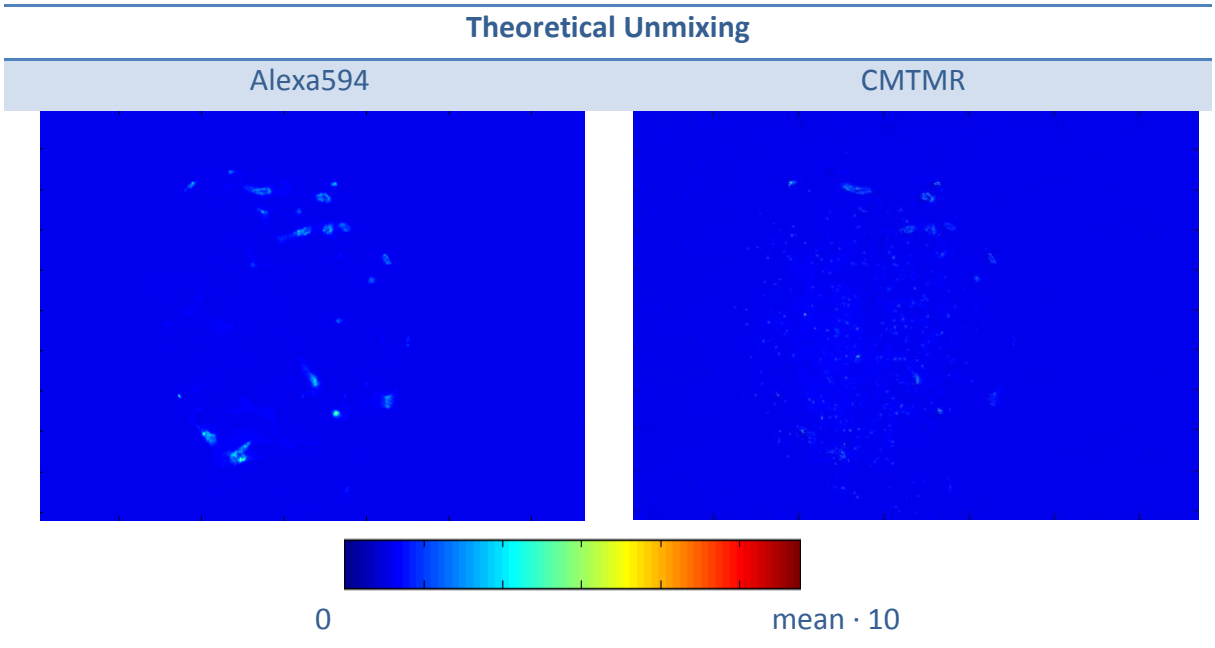


Figure 13. Unmixed images from theoretical method.

After obtaining the unmixed image for each fluorophore, both images were compared with the original ones, which were extracted from the images taken separately for each dye, as mentioned previously. The results of the analysis are presented below:

Method	CMTMR + Alexa594		CMTMR+CMTPX	
	Fluorophore	Error	Fluorophore	Error
Experimental	CMTMR	0.5735	CMTMR	3.8529
	Alexa594	1.5132	CMTPX	0.8972
Theoretical	CMTMR	0.0974	CMTMR	0.4110
	Alexa594	0.3734	CMTPX	0.9111

Table 2. Errors from real images unmixing.

As one can see, error coefficients for theoretical procedure are quite lower compared to those for experimental one. Values in theoretical method were taken from data sheets of the dye, while dealing with experimental method the values were estimated from the images, introducing then a significant amount of error. This error was reduced by the usage of a mask, which permits to select the more relevant values for the unmixing. Since it highly depends on how the mask is delimited and placed, the experimental method is not as accurate as the theoretical one is. However, the experimental method seems an interesting solution when previous information about the dyes is unavailable.

Looking at the results for CMTMR and CMTPX mixed images, the error coefficients resulted higher than those of CMTMR and Alexa594 did, especially on the experimental method, which might happen because both fluorescent dyes were labeling the same structures and both have similar spectra.

On the other hand, the method was also checked using the "synthetic" images. Error results show clearly the cause of the worse efficacy of the experimental procedure and its dependence on the selection and placement of the mask.

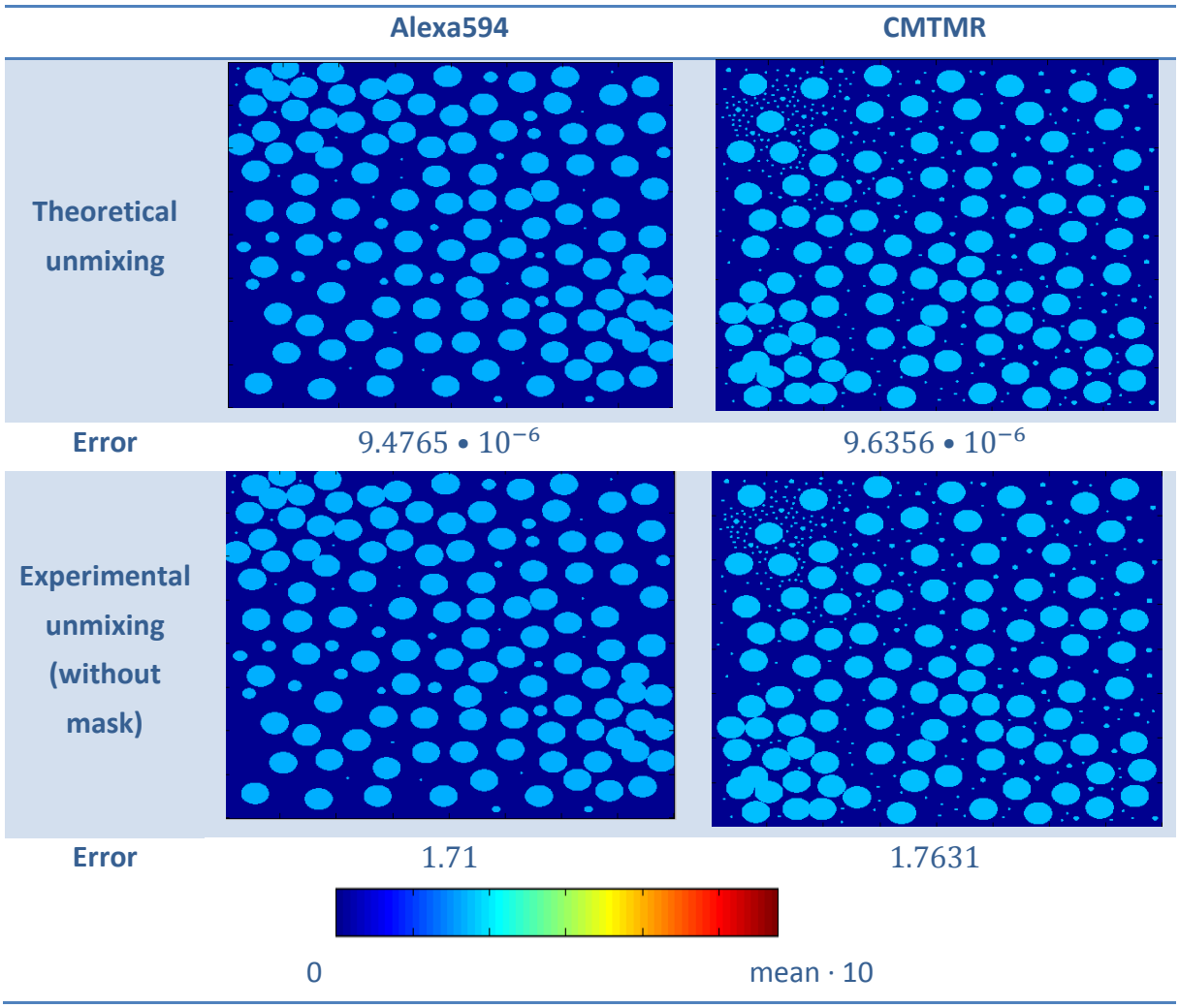


Figure 14. Results from synthetic images unmixing.

Since the experimental procedure's error depends highly on the mask, the attention was focused on the theoretical one. In order to check how the system behaved with the presence of noise, Gaussian white noise was introduced in the mixed synthetic images and then they were unmixed in the same way as before. The result is shown on figure 15.

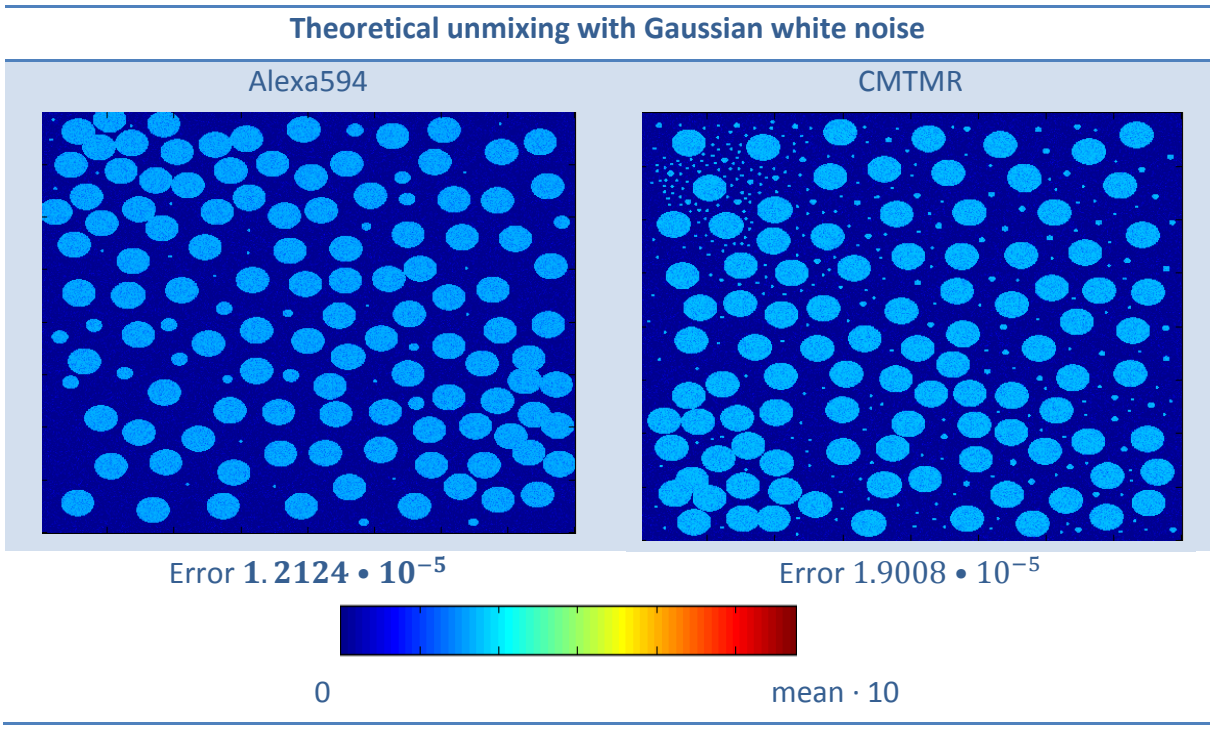


Figure 15. Results from synthetic images unmixing with presence of noise.

The inserted noise incremented insignificantly the error, demonstrating the robustness of the system, which still unmixed properly, unaware of the existing noise.

On the other hand, while calculating the pseudoinverse matrix using Matlab function $pinv$ or Singular Value Decomposition (SVD), the resultant error was the same for every case. About SVD procedure, the possibility of introducing a regularization parameter was useless, since in the present case only two fluorophores were mixed and the regularization parameter could choose only between two values: the lower value on matrix S or zero.

7 Conclusions and future lines

In the present work, a spectral unmixing procedure has been implemented in Matlab programming environment, for its specific usage in selective plane illumination microscopy (SPIM) images. SPIM has lately shown to have many advantages, such as high speed acquisition or reduced photobleaching, along with an extensive library of fluorescent stains, which makes necessary to achieve the correct detection and separation of multiples stains, existing in one image.

At the beginning of this study, commonly used algorithms for spectral unmixing were reviewed. Regarding it is a specific task, traditional methods for multispectral images have shown to work inefficiently when dealing with a pair of dyes. This may happen probably because they work without previous information about the data set, in an unsupervised manner. Then, it seems necessary to manually modify it, or become supervised. For example, algorithms like N-FINDR or SGA do not consider previous knowledge about the differences between fluorophores or about how they interact in each image. These methods exclusively look for the endmembers that mathematically are more appropriate. Then, while using SGA method it was frequently to observe how SGA separated one fluorophore into two endmembers, giving a wrong result at the end. Summing up, traditionally used algorithms are complex and because of that spend more time, being not as exact as it should be, and need human supervision and modification.

On the other hand, the weight matrix procedure uses a priori information: about the fluorophores and measurement methods, such as wavelengths and band pass filters. In this way, this procedure suggests to be more specific, more simple and straightforward. Furthermore, it has shown to a robust procedure, being less sensitive to noise and providing quite valid results in many circumstances.

This weight matrix may be built using already available data about the fluorophores, either from theoretical data set or from experimental sources, i.e., images presenting one dye separately where its intensity can be extracted. As seen above, experimental

procedure can be affected from multiple errors and noise. Obviously, theoretical method is not influenced by any external noise, obtaining better, more reliable results.

For future lines, work could be focused to improve its functioning and its internal operations. For example, the usage of SVD to calculate the pseudoinverse and the introduction of a regularization parameter could be studied when unmixing multispectral SPIM images with more than two dyes. This regularization would yield a better fitting and final unmixing result, which is totally recommended if the finest measurements are expected. Moreover, it would be highly regarded to extend its functionalities to other possibilities, and mainly to achieve its application to a wider variety of dyes as well as a more automatic usage of the software and gathering of data.

Furthermore, a future implementation of the proposed unmixing procedure could be done on other programming environment such as LabView, for a more visual and intuitive usage of the software.

8 References

[1] G. Martín Hernández. Design and implementation of new methods for spatial preprocessing prior to spectral unmixing of remotely sensed hyperspectral data. Doctoral thesis, University of Extremadura, 2013.

[2] N. Keshava and J. F. Mustard. *Spectral Unmixing*. IEEE SIGNAL PROCESSING MAGAZINE, pp. 44-57. January 2002.

[3] P.J. Martínez, R. M. Pérez, A. Plaza, P. L. Aguilar, M. C. Cantero and J. Plaza. *Endmember extraction algorithms from hyperspectral images*. ANNALS OF GEOPHYSICS, Vol. 49, No. 1, February 2006.

[4] Plaza and C.-I Chang. *An improved N-FINDR algorithm in implementation*. Conference on Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XI, SPIE Symposium on Defense and Security, SPIE Vol. 5806, Orlando, Florida, March 2005.

[5] D. C. Heinz and C.-I Chang. *Fully Constrained Least Squares Linear Spectral Mixture Analysis Method for Material Quantification in Hyperspectral Imagery*. IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, Vol. 39, No. 3, March 2001

[6] H.-xia Luo, J. Gong and J. Pan. *Effects of unsupervised fully constrained least squares linear spectral mixture analysis method on automatic classification of TM image*. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 34, Part XXX.

[7] C.-I Chang and D. Heinz. *Subpixel spectral detection for remotely sensed images*. IEEE Trans. Geosci. Remote Sensing, vol. 38, pp. 1144–1159, May 2000.

[8] N. Keshava. *A Survey of Spectral Unmixing Algorithms*. Lincoln Laboratory Journal, Vol. 14, No. 1, 2003.

[9] T. Pengo, A. Muñoz-Barrutía and C. Ortiz-de-Solórzano. *Spectral unmixing of multiply stained fluorescence samples*. Microscopy: Science, Technology, Applications and Education (FORMATEX), 2010.

[10] F. Chaudhry, C.-Cheng Wu, W. Liu, C.-I Chang and A. Plaza. *Pixel purity index-based algorithms for endmember extraction from hyperspectral imagery*. Recent Advances in Hyperspectral Signal and Image Processing, 2006.

[11] C.-I Chang and A. Plaza. *A Fast Iterative Algorithm for Implementation of Pixel Purity Index*. IEEE GEOSCIENCE AND REMOTE SENSING LETTERS, Vol. 3, No. 1, pp. 63-67, January 2006.

[12] C.-I Chang, C.-Cheng Wu, W.-min Liu and Y.-Chieh Ouyang. *A New Growing Method for Simplex-Based Endmember Extraction Algorithm*. IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, Vol. 44, No. 10, pp. 2804- 2819. October 2006.

[13] M. Simantiraki, R. Favicchio, S. Psycharakis, G. Zacharakis and J. Ripoll. MULTISPECTRAL UNMIXING OF FLUORESCENCE MOLECULAR TOMOGRAPHY DATA. Journal of Innovative Optical Health. Vol. 2, No. 4, pp. 353–364 (2009).

[14] M. E. Dickinson and M. W. Davidson. *Introduction to spectral imaging and linear unmixing*.
<http://zeiss-campus.magnet.fsu.edu/articles/spectralimaging/introduction.html>

[15] C.-I Chang. *Hyperspectral Data Processing: Algorithm Design and Analysis*. University of Maryland, Baltimore County (UMBC), Maryland, USA. April 2013.

9 Budget of the project

A detailed estimation of the costs of the project will be described. These costs are divided in two categories:

- Personnel costs involved in the project
- Costs of software, hardware and other costs.

Costs are expressed in Euros. Used Software tools and equipment are included in the final result.

- **Personnel costs**

In order to calculate the personnel costs, the people involved in the project as well as their role will be specified below. The cost per hour for each person is defined by an approximate value of the actual salary depending on their role. In table 3, the name of each participant, its category, the cost per hour, the total number of hours dedicated to the project and the final cost are shown. Jorge Ripoll was the supervisor and director of the project. Jun Abe was a professor and researcher from University of Bern, who provided microscopy images, then used for the development of the project.

Name	Category	Total hours	Euros/h	Final Cost
Jorge Ripoll	Professor	30h	35€	1.050€
Jun Abe	Researcher	60h	35€	2.100€
Guillermo Lozano	Junior Engineer	280h	20€	5.600€
Total Cost				8.750 €

Table 3. Personnel costs.

- **Software, Hardware and Other Costs**

The cost of the different materials and samples used, computers to analyze the data (and licenses) are included in this subsection. In table 4, the product used, its costs, and the number of days that the product was employed used are presented.

Product	Price	Use of the product	Final Cost for the project
Measuring system	100€/day	30 days	3000€
MATLAB license	600€	-	600€
MS Office license	200€		200€
Computer	600€	-	600€
Materials	20€	-	20€
Total Cost			4.420€

Table 4. Detailed software, hardware and material costs.

- **Total Cost of the Project**

Finally, the total sum of the costs of the different categories of the project is indicated in table 5.

Description	Total Cost
Personnel Costs	8.750€
Software and Hardware Costs	4.400€
Materials Costs	20€
Total Cost	13.170€

Table 5. Total cost of the project.

10 Appendix

Matlab codes

- Spectra selection

```
function [Emm,Abs] = spectra_Alexa594(w_em,w_ab)
% ** Values tabulated from Life Technologies
% ------------------------------------------------------------
% [Emm,Abs] = spectra_AL594(wavelength);
% --
% Input:
% ------
% a wavelength....| Wavelength at which to measure [nm]
%
% Output:
% -------
% Emm,Abs .........| Emission, Absorption spectrum [a.u.]
% ------------------------------------------------------------
[wav_em,emm] = AL594_em;
[wav_abs,abs] = AL594_abs;

Emm = interp1(wav_em,emm/100,w_em,'PCHIP');
Abs = interp1(wav_abs,abs/100,w_ab,'PCHIP');

return

% -------------------------------------
function [wav,em]=AL594_em
G = [
580 6.159652
581 6.657252
582 7.393682
583 8.439029
584 9.697656
...  ...
744 3.864513
745 3.820215
746 3.751669
747 3.572405
748 3.574366
749 3.524642
750 3.403642

];
```

```
% ---
wav = G(:,1); em = G(:,2);
% ---
wavmax = 745;
imax = find(wav>=wavmax);
P = polyfit(G(imax,1),log(G(imax,2)),1);
wav_end = (max(wav)+1):5:900;
G_end = exp(polyval(P,wav_end));
wav = [wav;wav_end'];
em = [em;G_end'];
% --
wavmin = 485;
imin = find(wav<=wavmin);
P = polyfit(G(imin,1),log(G(imin,2)+1E-3),1);
wav_ini = 400:5:(min(wav)-1);
G_ini = exp(polyval(P,wav_ini))-1E-3;
wav = [wav_ini';wav];
em = [G_ini';em];
em(em<0)= 0;

% -----------------------------------
function [wav,abs]=AL594_abs
G = [
300 19.88025
301 17.95591
302 16.22282
303 14.71552
304 13.54185
...  ...
668 0.6493893
669 0.7441167
670 0.6851355
671 0.7387549
672 0.6904975
673 0.7327971
674 0.5773011
675 0.7083706
];
% ---
wav = G(:,1); abs = G(:,2);
% ---
wavmax = 670;
imax = find(wav>=wavmax);
P = polyfit(G(imax,1),log(G(imax,2)),1);
wav_end = (max(wav)+1):5:900;
G_end = exp(polyval(P,wav_end));
wav = [wav;wav_end'];
abs = [abs;G_end'];
% --
```

```
wavmin = 305;
imin = find(wav<=wavmin);
P = polyfit(G(imin,1),log(G(imin,2)+1E-3),1);
wav_ini = 400:5:(min(wav)-1);
G_ini = exp(polyval(P,wav_ini))-1E-3;
wav = [wav_ini';wav];
abs = [G_ini';abs];
abs(abs<0)= 0;
```

- Weight Matrix Method

The following code was used for the specific case of Alexa594 and CMTMR fluorophores, then for another combination of dyes, some characteristics might be changed.

```
% - - - - - - - Input variables - - - - - - - - - - - - -
% 'Pairs_exc_emm_demm' - [excitation emission diameter]
% 'Fluorophores' -  contains the names of the fluorophores
% 'folder' - .tiff images main folder's pathname
% 'maskfile' - pathname of the mask
%
% - - - - - - - Output variables - - - - - - - - - - - -
% 'Iunmix' - unmixed images (:,:,Nfluo)
%

Pairs_exc_emm_demm = [561, 593, 46; ...
                      561, 628, 40; ...
                      561, 670, 30; ...
                      593, 628, 40; ...
                      593, 670, 30];

Fluorophores = {'Alexa594','CMTMR'};

folder = 'D:\...\20140725_SpectralUnmixing_AL594-CMTPX-
CMTMR\CMTMR+Alexa594';

maskfile = 'D:\...\20140725_SpectralUnmixing_AL594-CMTPX-
CMTMR\MASK.tif';

Nmeas = size(Pairs_exc_emm_demm,1);
Nfluo = length(Fluorophores);

%% Finding EXPERIMENTAL Weight Matrix
% M is a matrix of dimensions Nmeas x Nfluo

Umask = imread(maskfile);

for ii=1:Nfluo
    fluo_folder = [fileparts(folder) '\' Fluorophores{ii}];
```

```
    for im=1:Nmeas
        exc = num2str(Pairs_exc_emm_demm(im,1));
        emm = num2str(Pairs_exc_emm_demm(im,2));

        dd = dir([fluo_folder '\*' exc '*' emm '*.tif']);
        filename = [fluo_folder '\' dd(1).name];

        disp([exc '; ' emm ' > ' filename]);

        Iback = imread(filename,1);
        I = imread(filename,2);
        U = double(I);% - double(Iback);
        M(im, ii) = mean(U(Umask>0));
    end
end

Minv = pinv(M);


%% Finding THEORETICAL Weight Matrix
%M is a matrix of Nmeas x Nfluo dimensions
M=zeros(Nmeas,Nfluo);
 for ii=1:Nfluo

        for im=1:Nmeas
            %Select emission bandwidth and excitation wavelength
            wav_emm=(Pairs_exc_emm_demm(im,2)-
(Pairs_exc_emm_demm(im,3)/2):
Pairs_exc_emm_demm(im,2)+(Pairs_exc_emm_demm(im,3)/2));
            wav_exc=Pairs_exc_emm_demm(im,1);

            %Find fluorosphore spectrum data


fin_spec=str2func(strcat('spectra_',Fluorophores{ii}));

            %Use funcitons such as spectra_Alexa594 or
spectra_CMTMR to
            %find intensity values

            [Emm,Exc] = fin_spec (wav_emm,wav_exc);

            %Introduce Factor into Weight Matrix
            M(im,ii)=mean(Exc*Emm(:));
        end
end
Minv = pinv(M);
```

```
%% READING DATA
Idata = zeros(Nmeas,numel(U));

for im=1:Nmeas
    exc = num2str(Pairs_exc_emm_demm(im,1));
    emm = num2str(Pairs_exc_emm_demm(im,2));

    dd = dir([folder '\*' exc '*' emm '*.tif']);
    filename = [folder '\' dd(1).name];

    disp([exc '; ' emm ' > ' filename]);

    Iback = imread(filename,1);
    I = imread(filename,2);
    Idata(im,:) = I(:);
end

Iunmix = Minv*Idata;
Iunmix = permute(Iunmix,[2 1]);
Iunmix = reshape(Iunmix,size(I,1),size(I,2),Nfluo);

Iunmix(~isfinite(Iunmix)) = 0;
Iunmix(Iunmix<0) = 0;
```

Emission and excitation spectra

Emission and excitation spectra of the used dyes from Life Technologies are shown below:

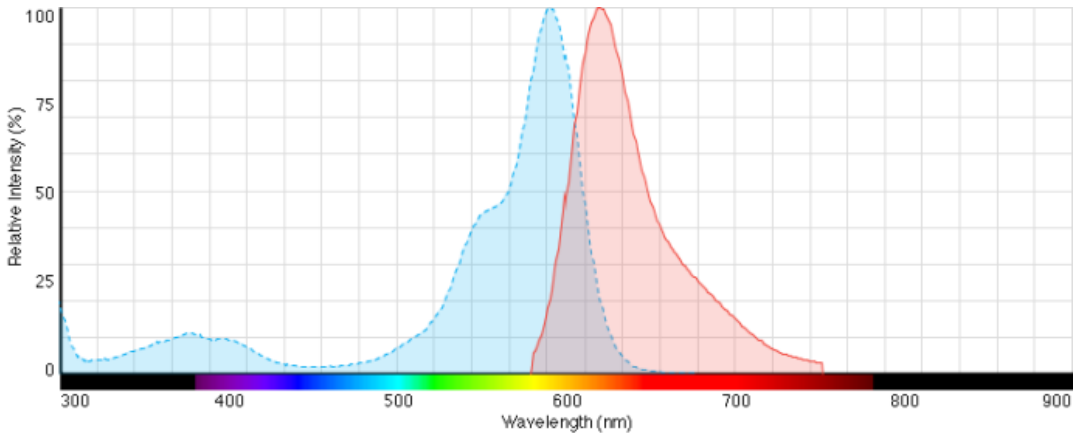


Figure 16. Alexa594 excitation (blue) and emission (red) spectra.

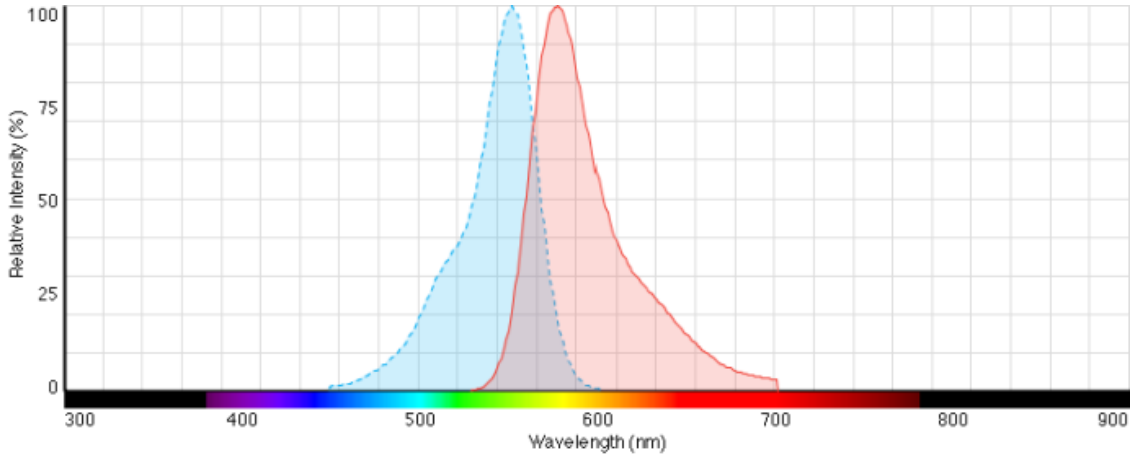


Figure 17. CellTracker Orange/CMTMR excitation (blue) and emission (red) spectra.

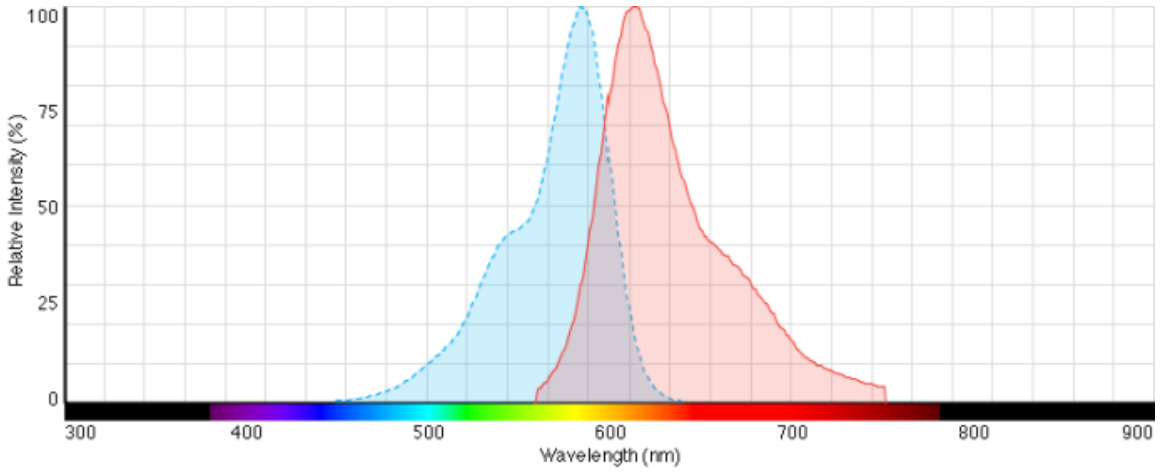


Figure 18. CellTracker Red/CMTPX excitation (blue) and emission (red) spectra.