Novel Reconstruction Errors for Saliency Detection in Hyperspectral Images*

Antonella Falini[1][0000−0001−6226−8967], Cristiano Tamborrino[2], Graziano Castellano[1], Francesca Mazzia[1][0000−0003−1072−9578], Rosa Maria Mininni[2][0000−0002−0230−1435], Annalisa Appice[1][0000−0001−9840−844X], and Donato Malerba[1][0000−0001−8432−4608]

[1] Department of Computer Science, University of Bari, Bari, Italy
{antonella.falini, francesca.mazzia, annalisa.appice, donato.malerba}@uniba.it
{g.castellano10@studenti.uniba.it}
[2] Department of Mathematics, University of Bari, Bari, Italy
{cristiano.tamborrino,rosamaria.mininni}@uniba.it

Abstract. When hyperspectral images are analyzed, a big amount of data, representing the reflectance at hundreds of wavelengths, needs to be processed. Hence, dimensionality reduction techniques are used to discard unnecessary information. In order to detect the so called "saliency", i.e., the relevant pixels, we propose a bottom-up approach based on three main ingredients: sparse non negative matrix factorization (SNMF), spatial and spectral functions to measure the reconstruction error between the input image and the reconstructed one and a final clustering technique. We introduce novel error functions and show some useful mathematical properties. The method is validated on hyperspectral images and compared with state-of-the-art different approaches.

Keywords: Error measures · Hyperspectral images · Saliency detection.

1 Introduction

HyperSpectral Imaging (HSI) is a technology that combines the imaging properties of a digital camera with the spectroscopic properties of a spectrometer able to detect the spectral attributes of each pixel in an image. Hyperspectral

* The research of Antonella Falini is founded by PON Project AIM 1852414 CUP H95G18000120006 ATT1. The research of Cristiano Tamborrino is funded by PON Project "Change Detection in Remote Sensing" CUP H94F18000270006. The research of the other authors is funded by PON Ricerca e Innovazione 2014-2020, the application of the proposed method to the saliency detection task is developed in partial fulfillment of the research objective of project RPASInAir "Sistemi Aeromobili a Pilotaggio Remoto nello spazio aereo non segregato per servizi" (ARS01 00820). While the application to the change detection task is developed in partial fulfillment of the research objective of the project "CLOSE to the Earth" (ARS01 00141).

sensors measure radiance, i.e. the radiant energy emitted, reflected, transmitted or received by a surface, by a large number of regularly spaced narrow bands covering a wide spectral range. All the information about reflectance across the whole spectral range of the sensor is contained by a single pixel, producing the so called *spectral signature*. Every spectral image can be thought as a cuboid: on the x- and y- directions, the spatial information is stored, while on the z-direction the spectral signature is smeared. Due to the size of the spectral signature, it is crucial to develop techniques able to reduce the dimensionality of the problem while keeping the important information. Characterizing what can be considered *salient* and what can be neglected becomes another challenging task. The concept of *saliency* refers to identifying parts, regions, objects or features that first draw visual attention and hence can be considered notable and important. Many attention models derive from the "Feature Integration Theory" [15], where it is stated which visual features can be considered important to drive human attention. In [7], Koch and Ullman propose a feed-forward model to combine these features and introduce the concept of a *saliency map* as a topographic map representing *conspicuousness* of scene locations. Saliency detection methods can be divided into top-down (see e.g., [4]), bottom-up (see e.g., [5]) approaches or a combination of both (see e.g, [18]). Besides being used as a technique to identify "what draws attention first", saliency detection can be employed also for more advanced tasks, as the so called "change detection". In this case it is relevant to highlight those pixels which result different when the hyperspectral image of the same area has been acquired at different time intervals, see for example [11] and references therein for a general overview.

To compress the dimension of the problem being able to extract relevant features, usually saliency detection methods are combined with matrix factorization techniques, like Principal Component Analysis (PCA) or Independent Component Analysis (ICA). In the present work, we use *Non Negative Matrix Factorization* (NMF) and the Gaussian Mixture Model clustering (GMM) to develop a saliency detection algorithm following a bottom-up approach. The main peculiarity of NMFs lies in factorizing a given matrix into the product of two matrices with non-negative values. Thanks to this property, NMFs are especially suitable for dealing with data which are only positive or null, like for instance intensities colors, and are preferable to other factorizations like PCA or ICA. NMFs were recently introduced in the context of saliency detection in [13], to discover the latent structure of image data-set, in [14] to extract features information from each superpixel, and in [9] to learn the basis functions from images patches. Once the factorization is performed, the reconstruction error is computed by using several functions. In particular, new error measures and their mathematical properties are analyzed. The introduction of such new functions provides a valuable contribution to output the sought saliency map. Then the reconstruction error vector, together with specific configurations of the available information is processed by the clustering algorithm in order to produce the final saliency map.

The remainder of the paper is organized as follows. In Section 2 the definitions of the employed error functions are provided. In Section 3 the structure of the proposed algorithm is presented. In Section 4 the experimental results are shown. In Section 5 final conclusions and possible directions for future work are discussed.

2 Reconstruction Error Measures

Every image can be represented as a tensor $\mathcal{A}$ of size $m \times n \times k$, where m and n identify the location of every pixel, while k represents the expression of the spectrum in the adopted spectral space. The input image $\mathcal{A} \in \mathbb{R}^{m \times n \times k}$ can be rearranged as a matrix A of size $p \times k$, with $p = m \times n$. In the resulting A, the spectral information of each pixel (x, y) is stored in a suitable row. The chosen factorization algorithm produces as output matrices $W \in \mathbb{R}_+^{p \times r}$ and $H \in \mathbb{R}_+^{r \times k}$ such that $A \approx Z := WH$. The number r is called *compression index*. The r columns of the basis matrix W can be thought as the latent factors embedded into the dataset. The elements of the coefficient matrix H can be considered as the weights associated to each factor.

We call *reconstruction error* the vector defined as $RE(A, Z)$ that collects the spectral differences between every pixel (x, y) in A and the pixels in the approximation Z. The RE can be thought as a dissimilarity index between the reconstructed image and the input one. In particular, since the background is the more extensive part of an image, we expect the NMF to reconstruct it quite accurately, discarding any other detail. Measuring RE should provide an image containing the salient part of the input A, therefore, finding a good error measure it is of fundamental importance. To this end we compute the following functions:

- *Euclidean distance* (ED), corresponds to the usual 2-norm for vectors and it is computed for every row of the matrix $A - Z$;
- *Spectral Angle Mapper (SAM)* [8], for every pixel (x, y) let ℓ denote its corresponding index in A and let $A_{\ell,:}$ be the ℓ-th row of the matrix A. SAM computes the angle between the spectral vectors $A_{\ell,:}$ and $Z_{\ell,:}$ in a space of dimension equal to the number of bands k.
- *Spatial Spectral Cross Correlation (SSCC)* [17], a measure of the similarity in the spatial-spectral domain of two images within a local window $V(\ell)$ centered at the specific pixel (x, y). The window $V(\ell)$ collects the indices $\ell_{i,j}$ corresponding to the pixels $(x + I, y + J)$ with $I, J = \pm d_V$ and $d_V \geq 1$.
- *The modified Z-score Index (ZID)* from [12],
- *SAMZID* [12], a combination of a modified Z-score Index and the SAM function,
- *SAM_g_ZID* we combine the SAM function with a new ZID index where a geometric mean has been used rather than the usual arithmetic mean:

$$SAM\_g\_ZID := [scale(\sin SAM)] \times [scale(ZID\_g)].$$

ZID_g is computed as the ZID indicator from [12] with the arithmetic mean replaced by μ_{g_ℓ}, the geometric mean for the vector $A_{\ell,:}$,

$$\mu_{g_\ell} := \left(\prod_{i=1}^{k} A_{\ell,i} \right)^{1/k}$$

The geometric mean describes the central tendency. The same mean μ_{g_ℓ} is used to compute the standard deviation for the vector $A_{\ell,:}$ required by the definition of the ZID.

– *Spatial Mean SAM (SMSAM)*, we consider the angles between the spectral vectors $A_{\ell,:}$ and $Z_{\ell,:}$ for the label ℓ varying in V and then we compute their arithmetic mean:

$$SMSAM(\ell) := \frac{1}{|V(\ell)|} \sum_{\ell_{i,j} \in V(\ell)} \arccos \left(\frac{A_{\ell_{i,j},:} Z_{\ell_{i,j},:}^{\top}}{\|A_{\ell_{i,j},:}\|_2 \|Z_{\ell_{i,j},:}\|_2} \right).$$

The symbol $|V(\ell)|$ denotes the number of elements of $V(\ell)$.

– *Spatial Mean Spectral Angle Deviation Mapper (SMSADM)*, we compute the arithmetic mean within a local window V of the deviation angles from their spectral mean inside V,

$$SMSADM(\ell) := \frac{1}{|V(\ell)|} \sum_{\ell_{i,j} \in V(\ell)} \arccos \left(\frac{E(A) E(Z)^{\top}}{\|E(A)\|_2 \|E(Z)\|_2} \right).$$

For a generic matrix S, the operator $E(S)$ is defined as:

$$E(S) := (S_{\ell_{i,j},:} - \mu(S)_{V(\ell)}).$$

The quantity $\mu(S)_{V(\ell)}$ denotes the spectral mean within the window $V(\ell)$ for the matrix S.

To improve the efficiency of processing every image, a superpixel segmentation can be applied to the original image A. We use the "simple linear iterative clustering" (`SLIC`) [16] algorithm which groups pixels in separate regions according to color. With this setting we compute three new error measures, *Spx-SSCC, Spx-SMSAM, Spx-SMSADM*, which respectively are the functions SSCC, SMSAM and SMSADM, where V is given as a tile produced by the superpixel algorithm.

2.1 Mathematical properties

In this section we show some useful properties for the proposed reconstruction errors.

It is straightforward to notice that $RE(A, Z) = RE(Z, A)$ for SAM, SSCC, SMSAM, SMSADM and their respective superpixel variants, i.e., the listed functions are *symmetric*.

Moreover, we can also prove that those error measures result *scale invariant*, i.e., $RE(\alpha A, \alpha Z) = RE(A, Z)$ for any $\alpha \in \mathbb{R}$. For the SAM measure we have,

$$RE(\alpha A, \alpha Z) = \arccos\left(\frac{\alpha A_{\ell,:}\alpha Z_{\ell,:}^{\top}}{\|\alpha A_{\ell,:}\|_2\|\alpha Z_{\ell,:}\|_2}\right) = RE(A, Z).$$

The same steps can be applied to *SMSADM*, *Spx-SMSADM* and to *ZID*. Regarding SSCC, SMSADM and their superpixel version we can observe that, given a generic matrix S and $\alpha \in \mathbb{R}$,

$$E(\alpha S) = \left(\alpha S_{\ell_{i,j},:} - \mu(\alpha S)_{V(\ell)}\right) = \left(\alpha S_{\ell_{i,j},:} - \alpha\mu(S)_{V(\ell)}\right) = \alpha E(S).$$

Then, for instance, if we use SSCC, for any $\alpha \in \mathbb{R}$ we have,

$$RE(\alpha A, \alpha Z) = \frac{\sum\limits_{\ell_{i,j}\in V(\ell)} E(\alpha A)E(\alpha Z)^{\top}}{\sqrt{\sum\limits_{\ell_{i,j}\in V(\ell)} \|E(\alpha A)\|^2}\sqrt{\sum\limits_{\ell_{i,j}\in V(\ell)} \|E(\alpha Z)\|^2}} = RE(A, Z)$$

All the proposed error measures are non negative besides *SAMZID* and *SAM_g_ZID*. Finally, we observe that any rotation or reflection applied in the spatial domain, would not change the relative location of the spectral vectors. Therefore, the RE value is the same up to the applied transformation.

3 Structure of the algorithm

The output of our algorithm will be a binary map where pixels belonging to the salient region have value 1 and 0 otherwise. The algorithm can be summarized by the following stages:

1- The input image is factorized by using one NMF algorithm provided by the open source Python library `Nimfa` [19], with compression index $r = 2$. NMF algorithms return as output a *basis* matrix W and a *mixture* matrix H such that $A \approx WH$.
2- The reconstruction error RE is computed by using several functions (see Section 2 for the details). Whenever a window V is considered, we set $d_V = \pm 2$.
3- A clustering algorithm is employed to classify the pixels between salient and not salient ones.
4- A post-processing step is performed to partly remove the noise in the final output. More precisely, for each pixel (x, y) a square window $V(\ell)$ with $d_V = \pm 4$ is considered. The same label as the 65% of the pixels within $V(\ell)$ is assigned to pixel (x, y) under consideration.

The set-up for parameters r and d_V has been empirically done by selecting the values which gave the best output.

In the following subsections, we recall some details and mathematical background.

3.1 NMF

Images datasets usually contain non negative elements. Therefore, when stored in matrices, in order to reduce the dimensionality of the problem, it is useful to apply first some non-negative factorization algorithm (NMF). In this study we use a variant of a standard NMF which is able to enforce sparseness constraints (SNMF) on the coefficient matrix H. Indeed, since we are trying to "compress" the spectral dimension, it is meaningful to set a null weight to certain coefficients.

We solve the following minimization problem

$$\min_{W\geq 0, H\geq 0} \frac{1}{2} \left\{ \|A - WH\|_F^2 + \eta \|W\|_F^2 + \beta \sum_{j=1}^{k} \|H(:,j)\|_1^2 \right\} \tag{1}$$

where $\| \|_F$ denotes the Frobenius norm, $\eta > 0$ is a parameter to minimize $\|W\|_F^2$ and $\beta > 0$ is a regularization parameter to balance the trade-off between the accuracy of the approximation and the sparseness of H. In particular, the initialization of the matrices W and H is done by averaging $m/5$ random columns and $n/5$ random rows of the matrix A. Then the descent gradient method is applied in order to compute the proper W and H that minimize the cost function (1). The problem (1) is a non convex optimization problem, therefore there is no guarantee to converge to a global minimum. Hence, we run the factorization 3 times and then, the matrices W and H with the lowest objective function value are chosen (parameter `n_run` $= 3$). We set $\eta = 1$, $\beta = 10^{-4}$ and, as a stopping criterion, the maximal number of iterations is given equal to 15.

3.2 Clustering stage

In order to classify the pixels belonging to the salient region we use the *Gaussian Mixture Model* (GMM) algorithm provided by the library `scikit-learn`, [10]. GMM is a "model-based" method as it relies on a probabilistic model for the data generation process. In particular, GMM assumes the points to be generated by a mixture of a finite number of Gaussian distributions. The number of components of our model is a-priori given, in particular, one cluster should gather salient pixels and another one should group the not-salient pixels, for a total of 2 clusters. At the beginning, each sample is assigned membership to the Gaussian it most probably belongs to, by computing posterior probability of each component. Then the *expectation maximization* (EM) strategy is applied to maximize the likelihood of the data, given those memberships.

As input to the GMM algorithm, for every error function, we provide the following configurations:

(1) `D_W`: the $RE(A, Z)$ and the basis matrix W.
(2) `D_WM`: the $RE(A, Z)$ and W_M where the index M denotes the column of W which outputs the maximal $\|RE(A, Z) - RE(A, W_{:,i}H_{i,:})\|_F$.
(3) `D_Wm`: the $RE(A, Z)$ and W_m where the index m denotes the column of W which outputs the minimal $\|RE(A, Z) - RE(A, W_{:,i}H_{i,:})\|_F$,

(4) `D_D-WM`: the $RE(A, Z)$ and $RE(A, W_M H_M)$.
(5) `D_D-Wm`: the $RE(A, Z)$ and $RE(A, W_m H_m)$.
(6) `D_D-W`: the $RE(A, Z)$, $RE(A, W_{:,1} H_{1,:})$ and $RE(A, W_{:,2} H_{2,:})$.

The GMM has been run with setting: the general covariance matrix is used for every component, the convergence threshold is set to 10^{-4}, k-means algorithm is used for three initializations. Moreover, in order to produce more robust results, we produce three different binary maps by running the GMM algorithm three times, where each new run starts with the last computed information. Supposing that the salient region covers a small part of the whole image, the best classification output is chosen to be the one which by far has two clusters of the same size.

4 Experimental Results

We test the proposed algorithm on the HS-SOD dataset[3] introduced in [6] and on the Hermiston dataset[4] which is a common benchmark for change detection task.

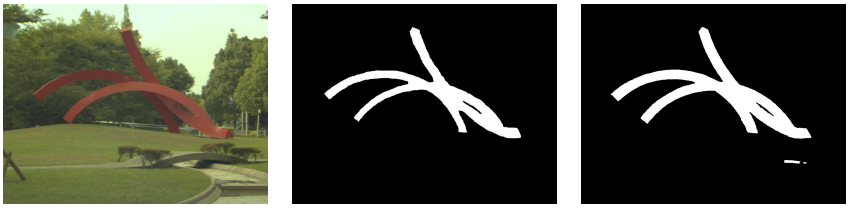
(a) image id= 26 (b) Ground Truth (c) SAMZID, `D_WM`, AUC:92.55%

Fig. 1. Output on image id = 26 from HS-SOD dataset.

4.1 HS-SOD dataset

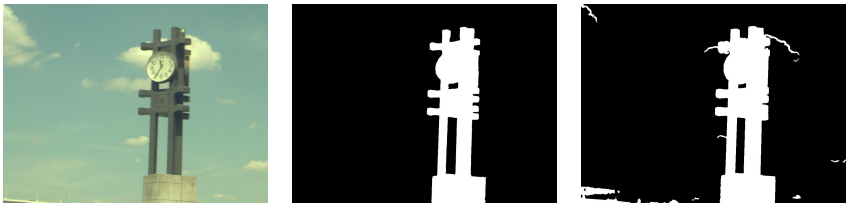
The HS-SOD dataset is a hyperspectral salient object detection dataset consisting of 60 hyperspectral images and their relative ground-truth binary images. There are 81 spectral bands, collected in the visible spectrum (380 - 780 nm). Each image has 1024×768 pixels resolution. The performance of the algorithm is evaluated by comparing the output saliency map with the provided ground-truth by using the Borji variant of the *Area Under the Roc Curve* (AUC) [2]. In the figures we show the output of the proposed method on four images, with respect

[3] https://github.com/gistairc/HS-SOD
[4] https://gitlab.citius.usc.es/hiperespectral/ChangeDetectionDataset/-/tree/master



(a) image id= 34 (b) Ground Truth (c) SMSAM, D_D-WM, AUC:95.48%

Fig. 2. Result on image id = 34 from HS-SOD dataset.



(a) image id= 2 (b) Ground Truth (c) SMSADM, D_D-W, AUC:92.80%

Fig. 3. Output of the algorithm on image id = 2 from HS-SOD dataset.

(a) image id= 38 (b) Ground Truth (c) SAM_g_ZID, D_D-W, AUC:60%

Fig. 4. The algorithm cannot fully identify the correct saliency map on image id = 38 from HS-SOD dataset.

to the best cluster configuration. For each image, in (a) we display the RGB rendering, in (b) the corresponding ground-truth and in (c) the best AUC obtained with the selected error measure and cluster configuration, see Fig. 1-2-3-4.

In Fig. 4 we also decided to show an example where our method partially fails to fully detect the correct salient region. This highlights one limitation of the proposed approach which is mainly driven by identifying spectral similarities using several error measures. When the spectral features of the salient region are alike to the ones of the background, our algorithm is not able to extract the correct information and the background is not distinguished at all from the salient object.

Table 1 reports the mean and standard deviation for the AUC on the whole dataset, obtained varying the error function. The label "Best" refers to the mean and standard deviation computed by taking into account only the best configuration for every image, regardless from the used error function.

In Table 2 we show the percentage of the images classified with AUC greater than 80% (class f_1), AUC between $65\% - 80\%$ (class f_2) and AUC less than 65% (class f_3) by varying the adopted error function and configuration. We observe that SMSAM has the highest percentage on f_1 for three different cluster configurations among all the error measures, while the Euclidean distance (ED) has the highest percentage on f_3 for all the cluster configurations. It is noteworthy that the obtained results are sensitive to the initialization of both the SNMF and the GMM algorithms. From the experiments we observed that by using the "Best" settings, 77% of the images were classified on class f_1, 22% on f_2 and only 2% on f_3. This result suggests the importance of defining an automatic selection algorithm for the suitable error measure.

Finally, in Table 3 we can see that the proposed algorithm, `Alg_1`, outperforms several techniques tested on HS-SOD. More in details, the competitors include: a preliminary version of the current approach, `Alg_0`, introduced by the same authors in [3]; a variant, named AISA, where an autoencoder architecture is constructed to elaborate HSI data [1], other methods analyzed in [6].

4.2 Hermiston dataset

The second dataset consists of two hyperspectral images of Hermiston City (Oregon) acquired by HYPERION sensor taken on year 2004 and 2009, respectively and a ground truth binary image. The hyperspectral images have 390×200 pixels resolution and 242 spectral bands. The proposed algorithm has been tested to detect the changes in the two pictures I_1, I_2 by determining the saliency map of the following input: $|I_1 - I_2|\,(|1 - I_{1s}/I_{2s}| + |1 - I_{2s}/I_{1s}|)/2$, where $I_{js} = I_j + s$, s is chosen such that the minimum value between I_1 and I_2 is 1 and the difference to 1 on the ratios has been applied in order to enhance the changes. From the results displayed on Fig. 5 we can see that the produced output and the expected ground truth are very much alike.

Table 1. Mean ± standard deviation for AUC-Borji computed on the HS-SOD dataset. In bold the maximum value achieved by every error measure.

RE	**AUC (mean ± std)**					
	D_W	D_WMax	D_Wm	D_D-WM	D_D-Wm	D_D-W
ED	**0.71** ± 0.11	0.70 ± 0.11	0.68 ± 0.12	0.68 ± 0.11	0.70 ± 0.11	0.69 ± 0.12
SAM	0.72 ± 0.12	0.74 ± 0.12	0.67 ± 0.12	**0.76** ± 0.11	0.74 ± 0.11	0.72 ± 0.13
SSCC	0.72 ± 0.13	0.72 ± 0.13	0.69 ± 0.13	0.73 ± 0.12	**0.74** ± 0.12	**0.74** ± 0.10
ZID	0.71 ± 0.12	0.73 ± 0.11	0.70 ± 0.12	**0.74** ± 0.12	0.73 ± 0.11	**0.74** ± 0.11
SAMZID	0.71 ± 0.12	0.71 ± 0.14	0.71 ± 0.13	0.73 ± 0.12	0.73 ± 0.12	**0.74** ± 0.11
SAM_g_ZID	0.72 ± 0.12	0.70 ± 0.14	0.70 ± 0.13	0.74 ± 0.12	0.74 ± 0.13	**0.75** ± 0.11
SMSAM	0.73 ± 0.12	0.75 ± 0.12	0.67 ± 0.12	**0.76** ± 0.12	**0.76** ± 0.12	0.71 ± 0.13
SMSADM	0.73 ± 0.12	0.73 ± 0.13	0.68 ± 0.13	0.73 ± 0.11	**0.75** ± 0.11	0.74 ± 0.11
Spx-SSCC	**0.73** ± 0.13	0.71 ± 0.13	0.68 ± 0.13	**0.73** ± 0.12	**0.73** ± 0.13	0.72 ± 0.12
Spx-SMSAM	0.72 ± 0.11	**0.74** ± 0.12	0.69 ± 0.13	0.73 ± 0.13	0.75 ± 0.12	0.73 ± 0.13
Spx-SMSADM	0.72 ± 0.11	0.70 ± 0.12	0.69 ± 0.13	**0.73** ± 0.13	0.72 ± 0.12	**0.73** ± 0.12
Best	**0.85 ± 0.08**					

Table 2. Percentage for the classes f_1, f_2 and f_3 for each error function and configuration. In bold the maximum value achieved according to the used cluster configuration.

RE	D_W			D_WM			D_Wm			D_D-WM			D_D-Wm			D_D-W		
	f_1	f_2	f_3	f_1	f_2	f_3	f_1	f_2	f_3	f_1	f_2	f_3	f_1	f_2	f_3	f_1	f_2	f_3
ED	23	38	38	22	35	43	18	32	50	18	32	50	23	33	43	20	35	45
SAM	28	37	35	38	32	30	20	28	52	37	43	20	30	43	27	32	32	37
SSCC	**33**	25	42	33	33	33	27	35	**38**	33	37	30	37	40	23	33	47	**20**
ZID	24	35	42	28	48	**23**	23	35	42	35	40	25	25	53	**22**	35	43	22
SAMZID	28	33	38	30	25	45	**35**	25	40	33	38	28	20	40	30	30	45	25
SAM_g_ZID	28	28	43	32	22	47	30	28	42	33	38	28	35	33	32	33	47	**20**
SMSAM	32	35	33	**43**	32	25	20	25	55	**43**	35	**22**	**42**	37	**22**	30	25	45
SMSADM	30	35	35	38	30	32	22	32	47	33	40	27	35	43	**22**	33	42	25
Spx-SSCC	32	35	33	32	37	32	18	35	47	37	33	30	40	28	32	28	42	30
Spx-SMSAM	25	43	**32**	42	30	28	22	35	43	35	33	32	38	35	27	30	35	35
Spx-SMSADM	25	37	38	30	27	43	22	33	45	38	28	33	35	32	33	**38**	32	30

Table 3. Comparison of the AUC-Borji with competitors from [3], [1] and [6]. In particular, these last ones are the following: Itti's method; a saliency detection method based on the computation of the euclidean distance (SED) and spectral angle distance (SAD); spectral group method (GS); orientation-based salient features method (OCM) in combination with: SED and GS (SOG), and, SED and SAD (SOS); the method using the spectral gradient contrast (SGC) computed by using superpixels with both, spatial and spectral gradients.

Method	Alg_1	Alg_0	AISA	Itti	SED	SAD	GS	SOG	SOS	SGC
AUC-Borji	0.8509	0.7971	0.7727	0.7694	0.6415	0.7521	0.7597	0.7863	0.8008	0.8205

Ground Truth | SAMSADM, D_D-WM, AUC:91.00% | ED, D_D-W, AUC:90.66%

Fig. 5. Results for the change detection task on Hermiston dataset.

5 Conclusions

In the present work we propose an algorithm to detect saliency regions in hyperspectral images by employing SNMF and several reconstruction errors based on spectral and spatial measures which add a notable contribution to better characterizing what should be understood as salient. This approach has been tested on HS-SOD and Hermiston datasets. The obtained results are promising, leave room to further investigations and to broader applications in data mining tasks such as semantic discovery patterns, collaborative filtering and recommender systems design. The output of our approach is strongly dependent on the initialization steps for the SNMF and for the clustering algorithm, and on the choice of the error measure as well. Hence a possible future direction could be to automatically identify the best settings. In particular, it would be interesting to study the relations between the adopted error function and the obtained output.

6 Acknowledgments

Antonella Falini, Cristiano Tamborrino and Francesca Mazzia are members of the INdAM Research group GNCS. Rosa Maria Mininni is member of the INdAM Research group GNAMPA. Annalisa Appice and Donato Malerba are members of the CINI Big Data Laboratory.

This work was also developed within the project "Modelli e metodi per l'analisi di dati complessi e voluminosi" of the University of Bari.

References

1. Appice, A., Lomuscio, F., Falini, A., Tamborrino, C., Mazzia, F., Malerba, D.: Saliency detection in hyperspectral images using autoencoder-based data reconstruction. In: International Symposium on Methodologies for Intelligent Systems, to appear. Springer (2020)
2. Borji, A., Cheng, M.M., Jiang, H., Li, J.: Salient object detection: A benchmark. IEEE transactions on image processing **24**(12), 5706–5722 (2015)

3. Falini, A., Castellano, G., Tamborrino, C., Mazzia, F., Mininni, R.M., Appice, A., Malerba, D.: Saliency Detection for Hyperspectral Images via Sparse-Non Negative-Matrix-Factorization and novel Distance Measures. In: 2020 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS). pp. 1–8. IEEE (2020)
4. Gao, D., Vasconcelos, N.: Discriminant saliency for visual recognition from cluttered scenes. In: Advances in neural information processing systems. pp. 481–488 (2005)
5. Han, J., Zhang, D., Hu, X., Guo, L., Ren, J., Wu, F.: Background prior-based salient object detection via deep reconstruction residual. IEEE Transactions on Circuits and Systems for Video Technology **25**(8), 1309–1321 (2014)
6. Imamoglu, N., Oishi, Y., Zhang, X., Ding, G., Fang, Y., Kouyama, T., Nakamura, R.: Hyperspectral Image Dataset for Benchmarking on Salient Object Detection. In: 0th International Conference on Quality of Multimedia Experience (QoMEX) (2018)
7. Koch, C., Ullman, S.: Shifts in selective visual attention: towards the underlying neural circuitry. In: Matters of intelligence, pp. 115–141. Springer (1987)
8. Kruse, F.A., Lefkoff, A., Boardman, J., Heidebrecht, K., Shapiro, A., Barloon, P., Goetz, A.: The spectral image processing system (SIPS)interactive visualization and analysis of imaging spectrometer data. Remote sensing of environment **44**(2-3), 145–163 (1993)
9. Liu, J., Liu, Y.: A model for saliency detection using nmfsc algorithm. In: International Conference on Computer Analysis of Images and Patterns. pp. 301–308. Springer (2009)
10. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
11. Radke, R.J., Andra, S., Al-Kofahi, O., Roysam, B.: Image change detection algorithms: a systematic survey. IEEE transactions on image processing **14**(3), 294–307 (2005)
12. Seydi, S.T., Hasanlou, M.: A new land-cover match-based change detection for hyperspectral imagery. European Journal of Remote Sensing **50**(1), 517–533 (2017)
13. Tang, J., Lewis, P.H.: Non-negative matrix factorisation for object class discovery and image auto-annotation. In: Proceedings of the 2008 international conference on Content-based image and video retrieval. pp. 105–112. ACM (2008)
14. Tao, D., Cheng, J., Song, M., Lin, X.: Manifold ranking-based matrix factorization for saliency detection. IEEE transactions on neural networks and learning systems **27**(6), 1122–1134 (2015)
15. Treisman, A.M., Gelade, G.: A feature-integration theory of attention. Cognitive psychology **12**(1), 97–136 (1980)
16. van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., Yu, T., the scikit-image contributors: scikit-image: image processing in Python. PeerJ **2**, e453 (6 2014)
17. Yang, Z., Mueller, R.: Spatial-spectral cross-correlation for change detection—a case study for citrus coverage change detection. In: ASPRS 2007 Annual Conference (2007)
18. Zheng, Q., Yu, S., You, X.: Coarse-to-fine salient object detection with low-rank matrix recovery. Neurocomputing (2019)
19. Zupan, B., et al.: Nimfa: A python library for nonnegative matrix factorization. Journal of Machine Learning Research **13**(Mar), 849–853 (2012)