# Dictionary Learning from Incomplete Data for Efficient Image Restoration

Valeriya Naumova
Section for Computing and Software,
Simula Research Laboratory,
Martin Linges 25, Fornebu, Norway
Email: valeriya@simula.no

Karin Schnass
Department of Mathematics,
University of Innsbruck,
Technikerstraße 13, 6020 Innsbruck, Austria,
Email: karin.schnass@uibk.ac.at

***Abstract*—In real-world image processing applications, the data is high dimensional but the amount of high-quality data needed to train the model is very limited. In this paper, we demonstrate applicability of a recently presented method for dictionary learning from incomplete data, the so-called Iterative Thresholding and $K$ residual Means for Masked data, to deal with high-dimensional data in an efficient way. In particular, the proposed algorithm incorporates a corruption model directly at the dictionary learning stage, also enabling reconstruction of the low-rank component again from corrupted signals. These modifications circumvent some difficulties associated with the efficient dictionary learning procedure in the presence of limited or incomplete data.**

**We choose an image inpainting problem as a guiding example, and further propose a procedure for automatic detection and reconstruction of the low-rank component from incomplete data and adaptive parameter selection for the sparse image reconstruction. We benchmark the efficacy and efficiency of our algorithm in terms of computing time and accuracy on colour, 3D medical, and hyperspectral images by comparing it to its dictionary learning counterparts.**

## I. Introduction

In signal and image processing, the most notable recent advances are based on the observation that natural (often high-dimensional) signals can be well approximated as a linear combination of a small (sparse) number of elementary signals, the so-called atoms, from a prescribed basis or frame, called dictionary. Formally, representing each signal as a column vector $y_n \in \mathbb{R}^d$ and arranging normalised atoms as columns of the dictionary $\Phi = (\phi_1, \dots, \phi_K) \in \mathbb{R}^{d\times K}$, the sparse model is described as

$$y_n = \Phi x_n \text{ and } \|x_n\|_0 \ll d. \tag{1}$$

In this expression, $x_n \in \mathbb{R}^K$ is the sparse representation of $y_n$ and the function $\|\cdot\|_0$ counts the number of non-zero entries of its argument. The problem of finding a sparse vector $x_n$ in (1) is referred to as sparse coding and is an NP-hard optimisation problem for overcomplete dictionaries, i.e., when $d < K$. In practice, greedy algorithms and convex relaxation alternatives are successfully employed to solve this problem. Thus, the main focus of several communities has recently been on automating choice of the dictionary $\Phi$. Now it is evident that even though some analytically-defined dictionaries such as Wavelet or Overcomplete Discrete Cosine Transform are fast and easy to implement, learning the dictionary from the given data for a specific task gives state-of-the-art results in many signal and image processing applications. There exist a multitude of dictionary learning algorithms to choose from, see [1]–[3], for instance.

Though successful, the dictionary learning problem has traditionally been restricted to problems when a large number of clean high-quality signals are available for training. In addition, due to the computational complexity of the problem, the majority of the learning algorithms have been restricted to work with relatively small signals. Recent works [4] tried to address the latter issue by observing that learned dictionaries have a certain sparse structure, which can be efficiently utilised. These novel algorithms are capable of handling signals at the dimension of several thousands, and learn on millions of data signals. At the same time, aspects related to data availability for the model training has mostly been ignored in the literature.

We tried to circumvent the problem of data availability and proposed an algorithm for learning dictionaries from incomplete/masked training data, the so-called *Iterative Thresholding and K residual Means for Masked data* (ITKrMM) [5]. Being an extension of the theoretically-justified and numerically efficient *Iterative Thresholding and K residual Means* algorithm, [6], the ITKrMM algorithm incorporates a signal corruption model into the dictionary learning phase and, additionally, allows to recover a low-rank component, again from incomplete data. Specifically, the corrupted data is described using the concept of mask $M$, given by a binary matrix, that is multiplied to the signal $y$ entry-wise and zeros in $M$ indicates the removed/missing data pixel.

In the current paper, we demonstrate the applicability of the ITKrMM algorithm to deal with high-dimensional data, e.g., colour and 3D images. We also introduce procedures for automatic selection of the low-rank component size and adaptive parameter choice for the sparse coding in image inpainting tasks, leading to state-of-the-art results in terms of the reconstruction quality and computational complexity.

The rest of the paper is organised as follows: Section II contains a brief description of the ITKrMM algorithm. An adaptation of the algorithm to deal with high-dimensional data as well as various simulation results that demonstrate the

ISBN 978-0-9928626-7-1 

effectiveness of the proposed algorithm for colour, 3D, and hyperspectral image inpainting are presented in Section III. Finally, we conclude the paper with a brief description of the contribution and open questions for future work.

We start with providing a standard notation used in the paper: for a matrix $A$, we denote its (conjugate) transpose by $A^\star$. We denote by $P(A)$ the orthogonal projection onto the column span of $A$, i.e. $P(A) = A(A^\star A)^{-1}A^\star$ and by $Q(A)$ the orthogonal projection onto the orthogonal complement of the column span of $A$, i.e. $Q(A) = \mathbb{I}_d - P(A)$, where $\mathbb{I}_d$ is the identity matrix in $\mathbb{R}^d$.The restriction of the dictionary $\Phi$ to the atoms indexed by $I$ is denoted by $\Phi_I$, i.e. $\Phi_I = (\phi_{i_1}, \dots, \phi_{i_S})$, $i_j \in I$.

## II. ITKrMM algorithm

The ITKrMM algorithm learns a dictionary $\Phi$ from corrupted signals $M_n y_n$ under the assumption that the signals $y_n$ are sparse in the dictionary $\Phi$ and allows to recover the low-rank component $\Gamma$ again from corrupted data. The algorithm belongs to the class of alternating projection algorithms, which alternate between sparsely approximating the signals in the current version of the dictionary and updating the dictionary based on the sparse approximations. For sparse approximation, ITKrMM uses thresholding and for the atoms update – residual averages and as such has the advantage of being computationally light and sequential.

We start with providing some intuition behind the ITKrMM algorithm before presenting the algorithm itself and refer to [5] for a detailed description of the algorithm. Given the corrupted signal $My$, its proper dictionary representation is obtained, based on the following considerations: firstly, as the signal's energy could be distorted by the mask, $\|My\|_2 \leq \|y\|_2$, we cannot assume that the corrupted signal is normalised. Moreover, the corrupted signal is not sparse in the dictionary $\Phi$ but rather in its corrupted version $M_n\Phi$, which is indeed not a dictionary as its columns are in general non-normalised, again because of $M$. Thus, the proper $My$ representation is

$$M_n y_n = \sum_{\substack{i\in I_n:\\ M_n\phi_i\neq 0}} x_i \|M_n\phi_i\|_2 \cdot \frac{M_n\phi_i}{\|M_n\phi_i\|_2},$$

where $I_n$ denotes the support. To recover $I_n$ via thresholding, we check the $S-$largest inner products between the corrupted signal and the renormalised corrupted atoms,

$$\begin{aligned} I_n^t &= \arg\max_{I:|I|=S} \sum_{\substack{i\in I:\\ M_n\phi_i\neq 0}} \frac{|\langle M_n\phi_i, M_n y_n\rangle|}{\|M_n\phi_i\|_2} \\ &= \arg\max_{I:|I|=S} \sum_{i\in I} \|P(M_n\phi_i)M_n y_n\|_2. \end{aligned}$$

Given the generating support $I_n^t$, the dictionary update rule is performed by calculating for each atom residual means over the corrupted signals and then by rescaling each atom according to the number of times it has been observed.

Finally, as most of the natural signals are not perfectly sparse but rather modelled as the orthogonal sum of a low-rank and a sparse component, recovery of the low-rank component has to be addressed. Otherwise, we risk to end up with an ill-conditioned and coherent dictionary, where most atoms are distorted towards the low-rank component. For clean signals the removal of the reconstructed low-rank component $\tilde\Gamma$ is quite straightforward, but the situation becomes more complicated for the corrupted signals as the mask destroys the orthogonality between the dictionary and the low-rank component. To overcome this difficulty, we incorporate into the dictionary update rule the projection on the estimated low-rank component and ensure that the output dictionary is orthogonal to the recovered component.

These considerations lead to the iterative algorithm, presented below. Interpreting all signals as 1-sparse in a dictionary of one atom, we can adjust the ITKrMM algorithm for the low-rank component recovery. In particular, a step with the sparse support recovery is omitted as majority of the training signals are expected to contain the one new atom. Additionally, we iteratively learn the low-rank component atom by atom.

**Algorithm II.1** (ITKrMM - one iteration)**.** *Given an estimate of the low-rank component $\tilde\Gamma$, an input dictionary $\Psi$ with $\Psi^\star\tilde\Gamma = 0$, a sparsity level $S$ and $N$ corrupted training signals $y_n^M = (M_n y_n, M_n)$ do:*

- *For all $n$ set $M_n\tilde y_n = Q(M_n\tilde\Gamma)M_n y_n$.*
- *For all $n$ find*

$$I_n^t = \arg\max_{I:|I|=S} \sum_{i\in I: M_n\phi_i\neq 0} \frac{|\langle M_n\phi_i, M_n\tilde y_n\rangle|}{\|M_n\phi_i\|_2}$$

.

- *For all $k$ calculate*

$$\bar\psi_k = \sum_{n:k\in I_n^t} \big[\mathbb{I}_d - P(M_n(\tilde\Gamma, \Psi_{I_n^t})) + P(M_n\psi_k)\big]M_n\tilde y_n \times \\ \times \operatorname{sign}(\langle\psi_k, M_n\tilde y_n\rangle) \qquad \textit{and} \qquad W_k = \sum_{n:k\in I_n^t} M_n.$$

- *Set $\bar{\tilde\psi}_k = Q(\tilde\Gamma)W_k^\dagger\bar\psi_k$ and output*

$$\bar\Psi = (\bar{\tilde\psi}_1/\|\bar{\tilde\psi}_1\|_2, \dots, \bar{\tilde\psi}_K/\|\bar{\tilde\psi}_K\|_2).$$

## III. Image inpainting with ITKrMM

Image inpainting is the process of filling in missing information in images. The areas with missing information can be either single pixels such as in noisy digital photos, larger continuous regions stemming from scratches, or other objects like text or date stamps on an image. Image inpainting has become an active field of research in the mathematical and engineering communities, and the methods and approaches vary greatly, especially for 2D problems. The dictionary-based inpainting can be formulated as recovering sparse coefficients $x_I$ such that every masked patch is sparse in the masked dictionary

$$My \approx M\Phi_I x_I \text{ for } |I| \leq S. \tag{2}$$

Then the reconstructed patch is given as $\tilde y \approx \Phi\tilde x_I$ with $\tilde x_I \approx x_I$. Our goal is to evaluate the robustness and simplicity of the ITKrMM algorithm for dictionary-based inpainting, and

ISBN 978-0-9928626-7-1 © EURASIP 2017

we therefore do not compare our method with state-of-the-art methods for image inpainting, but compare instead against other dictionary-based inpainting algorithms like the weighted KSVD (wKSVD) [8].

In [5] we demonstrated that the ITKrMM achieved optimal performance for grayscale image inpainting in standard benchmark tests. We here expand on the previous method and introduce several improvements, also for application to colour, 3D images, and hyperspectral images. Our previous results indicated that the size of the low-rank component played an important role in the performance of the ITKrMM algorithm. In particular, we observed that for grayscale images with low corruption levels, the low-rank component of higher size led to better inpainting results. To account for this adaptively, we propose an automatic determination of the low-rank component size. Since the ITKrMM algorithm is iterative, it provides the possibility to determine the size of the low-rank component, specifically, by evaluating the ratio between the energies captured by the last low-rank atom $\gamma_l$, i.e. $\sum_n \|P(M_n\gamma_l)M_n y_n\|_2^2$ to the signal energy expected to be captured by a dictionary atom $\frac{1}{K}\sum_n \|M_n y_n\|_2^2$. As soon as the ratio between these energies stabilises, we stop adding low-rank atoms and turn to the dictionary learning step.

Assuming we have a learned dictionary, we still need to approximate the sparse coefficients $x_I$. In [5] we introduced the masked version of the Orthogonal Matching Pursuit (OMP) to solve (2), which is adapted towards damaged dictionaries by including additional normalisation step. In the masked OMP, the number of non-zero components in $x_I$ was fixed a priori, depending on the corruption nature. As the sparsity level might vary from patch to patch, depending on the local image structure and corruptions, an adaptive selection of the sparsity level for each patch could lead to a better reconstruction quality and also be more time efficient, esp. for smooth/homogenous images. Thus, we introduce an adaptive masked OMP where the sparsity level is chosen according to the specified deviation $\epsilon$ in the representation, i.e. $\|My - M\Phi x_I\|_2 \leq \epsilon$ such that $|I| \leq S_{\max}$, where $S_{\max}$ depends on the patch dimension. In [8] the authors argued that the error-based stopping criteria in OMP for colour image inpainting should be substituted by a fixed number of OMP iterations, or prescribed sparsity level to speed up the process. On the contrary, we observe that the masked OMP equipped with the error-stopping rule leads to better and faster results as the ITKrMM-learned dictionaries are in general flat and, thus, on average patches have sparser representation compared to an a priori prescribed value.

### A. Colour image inpainting

In Figure 1 we present a classical example of text removal, which has been used in several previous studies with reports of state-of-the art performance of 32.45 dB using the wKSVD algorithm [8]. Using the ITKrMM algorithm with automatic choice of the low-rank component size $L$ and adaptive masked OMP, we achieve the performance of 39.68 dB with patches of size $10 \times 10 \times 3$ and the redundant dictionary of size $K = 600 - L$. For a fair comparison, it is worthwhile to mention that the sparsity level for wKSVD-based inpainting is also chosen adaptively, leading to a better performance of 35.12 dB than it was reported in the original paper [8]. Let us now describe our setup for dictionary-based inpainting in more details.

**Data:** We consider colour images from a standard image database with 30, 50 or 70% of randomly erased pixels. A mask is applied to the whole image such that missing pixels may not be the same in each colour channel. To speed up computation, all images are rescaled to have a standard height of 256 pixels, keeping the original aspect ratio. We then extract all possible patches of size $p \times p \times 3$ (containing the RGB layers) pixels from the corrupted image and the corresponding mask. We consider patches of size $p = 8$ and $p = 10$ to compare performance of the algorithms. The vectorised corrupted patch/mask pairs are then given to the dictionary learning algorithms.

**Dictionary & low-rank component:** The ITKrMM-learned dictionaries are of size $K = 2d - L$ atoms, where $d = 3p^2$ is the dimension of the patches and the size of the low-rank component $L$ is chosen automatically according to the iterative procedure described above and we stop as soon as the ratio between the energies does not decrease by more than 20% from the previous iteration. wKSVD-learned dictionaries are of size $K = 2d$, where the first atom is constant, i.e. $\phi_1 = c$, as suggested in the original paper, this corresponds to $L = 1$ in the ITKrMM case.

**Initialisation & sparsity level:** The same initial dictionary, consisting of orthonormal random vectors, are used for both algorithms. The dictionaries are trained using $S = p - L$ coefficients per atom, with $L = 1$ for wKSVD. We run 20 iterations for the low-rank component learning and all available patch/mask pairs; whereas for the dictionary learning we run 80 iterations on all available patch/mask pairs for both algorithms. Note that wKSVD-based algorithms are suggested to be pre-learned on an image database and then adapted to image patches at hand [8]. In our case, we skipped this step, but still observe a good performance of the algorithm. The procedure for automatic $L$ choice confirms our previous observations that the image nature and the corruption level determines the size of the low-rank component. $L = 3$ was chosen for more textured images like Mandrill and low corruption levels, whereas the algorithm picked up $L = 1$ or $L = 2$ for 70% corruptions and/or smooth images like Castle.

**Dictionary-based inpainting:** We first reconstruct every damaged image patch using the adaptive masked OMP with $S_{\max} = d/3$ and $\epsilon = 10^{-6}$ as the stopping criteria. Afterwards we reconstruct the full image by averaging every pixel over all patches in which it was contained.

**Comparison/Error:** We measure the success of recovery for the different pairs of dictionaries by the peak signal-to-noise ratio (PSNR) between the original image $Y$ and the recovered version $\tilde{Y}$. For $Y, \tilde{Y}$ both images of size $d_1 \times d_2$

ISBN 978-0-9928626-7-1 © EURASIP 2017

the PSNR is defined as

$$\text{PSNR in dB} = \log_{10}\left(\frac{(\max_{i,j} Y(i,j) - \min_{i,j} Y(i,j))^2}{\frac{1}{d_1 d_2}\sum_{i,j}(Y(i,j) - \check{Y}(i,j))}\right).$$

For each corruption level, the results in Table I are averaged over 5 runs, each with a different mask and initialisation, to account for the variability of the PSNR for different mask realisations.

Similarly, as observed for grayscale images in our previous paper, Table I shows that both algorithms perform about equally well, although the ITKrMM performs slightly better for more textured images (Barbara, Mandrill) and smaller corruption level in general, whereas for the higher corruptions wKSVD is slightly better. A plausible explanation is that the ITKrMM algorithm learns better more textured (high frequency) atoms, while wKSVD produces smooth (low frequency) atoms. For higher corruption levels, the learned ITKrMM dictionaries are simply noisy versions of their counterparts learned from less corrupted data, whereas the wKSVD dictionaries mainly contain noisy versions of the smooth atoms from their 30% counterparts and not many recognisable copies of high-frequency atoms. At the same time, we have not observed any differences in colour recovery for both algorithms.

Comparing compute time of both algorithms, we observe that ITKrMM is about 10 times faster than wKSVD for patch size of $8 \times 8$ and about 13 times faster for patch size of $10 \times 10$[1]. In general the computational efficiency of ITKrMM over wKSVD becomes more pronounced for larger $d$ resp. $K$. We expect that the ITKrMM run time may be further significantly improved by parallelisation of the ITKrMM algorithm and learning the dictionary on a pre-selected subset of the representative signals. For the former, we expect a close-to linear speedup with the number of processors, whereas the latter issue we intend to investigate in future work.

| Corr. | Algorithm | Barb. | Castle | House | Man. | Pepp. |
|---|---|---|---|---|---|---|
| rand. 30% | *Noisy Im.* | *11.72* | *11.01* | *9.85* | *10.64* | *11.18* |
| | wKSVD | 39.24 | **42.56** | 40.93 | 33.48 | 39.53 |
| | ITKrMM | **40.40** | 42.18 | **41.63** | **34.92** | **40.55** |
| rand. 50% | *Noisy Im.* | *9.49* | *8.77* | *7.63* | *8.41* | *8.96* |
| | wKSVD | 35.70 | **37.46** | 38.18 | 29.73 | **36.27** |
| | ITKrMM | **36.09** | 36.52 | **38.33** | **30.71** | 36.17 |
| rand. 70% | *Noisy Im.* | *7.45* | *6.73* | *5.58* | *6.36* | *6.91* |
| | wKSVD | **30.20** | **28.97** | **31.81** | **25.37** | **29.49** |
| | ITKrMM | 29.22 | 27.77 | 31.74 | 24.51 | 28.74 |

TABLE I: Comparison of the PSNR (in dB) for inpainting of *Barbara, Castle, House, Mandrill,* and *Peppers,* images with various corruption levels based on dictionaries learned with wKSVD and ITKrMM on all available corrupted image patches of size $8 \times 8$. Note that the size of low-rank component as well as coefficients in masked OMP are adaptively selected in the ITKrMM algorithm.

[1]as observed by running both algorithms in unoptimised form on the UIBK computing cluster consisting of 45 nodes with 20 Intel Xeon (Haswell) computing cores each, all nodes equipped with 64GB RAM except for two nodes with 512GB RAM

### B. Inpainting of 3D images

To evaluate the performance of the ITKrMM algorithm for 3D image inpainting, we used synthetic cerebral MRI volumes available on BrainWeb [9] with 80% random missing voxels. As most of the settings for the ITKrMM-based inpainting are kept similar to the ones presented above, we describe briefly some introduced changes. From the MRI volume of size $217 \times 181 \times 181$ we extract patches of size $8 \times 8 \times 8$ from each 8th slice in $z$ direction. These pre-selected patches are chosen to maintain computational requirements reasonable. We learn the dictionary of size $K = 2d - L$, where $d = p^3 = 512$, using signals with the maximum allowed corruption level of 80%. For two realisations of the random mask, we run 20 iterations for the low-rank component learning and 80 iterations for dictionary learning on extracted patch/mask pairs. Figure 2 shows that even with such high corruption level, we can still see incredibly faint details in the restored anatomical structures.

### C. Inpainting of hypersspectral images

Finally, we illustrate a good performance of the ITKrMM algorithm for spatial recovery of hyperspectral data from Mars Observer [10]. Missing data are a standard problem in astronomy. Combined with its inherent high-dimensionality and complicated geometry, a data recovery/inpainting algorithm has to be fast and efficient. Figure 3 shows the recovery performance for varying degrees of missing voxels. The accurate recovery in combination with low computational requirements make the algorithm suitable for large scale data analyses such as commonly encountered in cosmology.

## IV. Conclusion

We have demonstrated the robustness of the ITKrMM algorithm for colour, 3D, and hyperspectral image inpainting. By introducing adaptivity in selection of the low-rank component and the sparsity level in the sparsity coding procedure, we further improve the performance and efficiency. The direct comparison with the state-of-the art algorithm wKSVD illustrated the noticeable improvements in terms of required computational resources and similar recovery results. The ITKrMM algorithm reproducing the results for grayscale and colour images is available at the Karin Schnass webpage.

Beyond the theoretical justification of the ITKrMM algorithm, we intend to introduce the concept of iterative dictionary relearning with adaptive selection of the training signals. This procedure could be beneficial for image inpainting with large amount of missing data as, for instance, frequently encountered in cosmic data. Potential advantage of our approach is that we do not need additional clean data for dictionary prelearning and, thus, the technique could be applied in various applications, where the amount and the quality of the data pose a big challenge. Another interesting future direction is related to the adaptive pre-selection of the signals for the dictionary learning, which is highly relevant and desirable in the case of high-dimensional data.

ISBN 978-0-9928626-7-1 © EURASIP 2017

(a) image with text

(b) wKSVD: PSNR is 35.12 dB

(c) ITKrMM: PSNR is 39.68 dB

Fig. 1: Inpainting example: *Text removal*

Fig. 2: Inpainting example for 3D images: A sagittal $(x, y)$ slice of the original synthetic MRI volume from BrainWeb [9] (left), its corrupted version with 80% randomly missing voxels (middle), and inpainted results with ITKrMM dictionary (right)

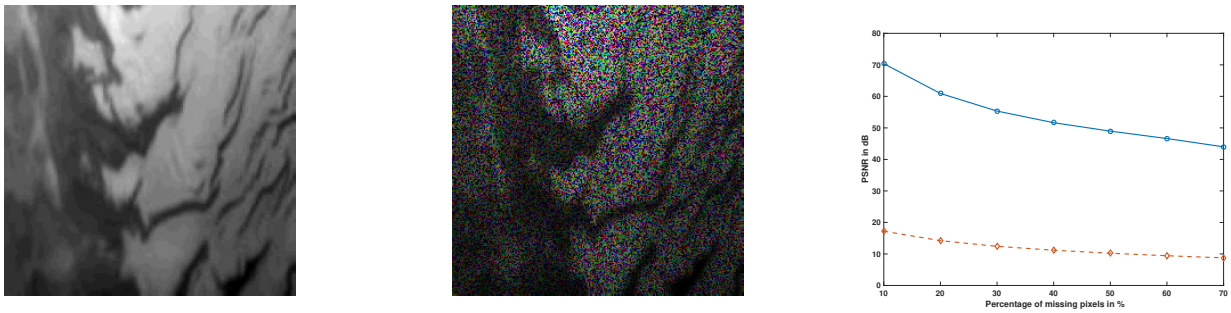


Fig. 3: Inpainting example for hyperspectral astronomical image: A spatial recovery of the Mars Express observation with ITKrMM dictionary (left) from its corrupted version with 50% randomly missing voxels (middle), and ITKrMM recovery performance for varying degrees of missing voxels: PSNR values for the ITKrMM-recovered images (blue solid line) and the corrupted image (red dotted line) (right)

## ACKNOWLEDGMENT

V. Naumova acknowledges the support of project No 251149/O70 'Function-driven Data Learning in High Dimension' (FunDaHD) funded by the Research Council of Norway and K. Schnass is in part supported by the Austrian Science Fund (FWF) under Grant no. Y760. In addition, part of the computational results has been achieved using the HPC infrastructure LEO of the University of Innsbruck.

## REFERENCES

[1] R. Rubinstein, A. Bruckstein, and M. Elad, Dictionaries for sparse representation modelling, *Proceedings of the IEEE,* vol. 98, no. 6, pp. 1045-1057, 2010.

[2] K. Schnass, A personal introduction to theoretical dictionary learning, *Internationale Mathematische Nachrichten,* vol. 228, pp. 5-15, 2015.

[3] J.-L. Starck, F. Murtagh, and J. Fadili, *Sparse image and signal processing*, Cambridge University Press, 2015.

[4] J. Sulam, B. Ophir, M. Zibulevsky and M. Elad, Trainlets: Dictionary Learning in High Dimensions, *IEEE Transactions on Signal Processing,* vol. 64, no. 12, pp. 3180-3193, 2016.

[5] V. Naumova and K. Schnass, Dictionary learning from incomplete data, *arXiv:1701.03655*, pp. 1-22, 2017.

[6] K. Schnass, Convergence radius and sample complexity of ITKM algorithms for dictionary learning, *accepted to Applied and Computational Harmonic Analysis*, 2016.

[7] M. Aharon, M. Elad, and A.M. Bruckstein, K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation, *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311-4322, 2006.

[8] J. Mairal, G. Sapiro, and M. Elad, Sparse representation for colour image restoration, *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 53-69, 2008.

[9] K. Remi, A. Evans, and G. Pike, MRI simulation-based evaluation of image-processing and classification methods, *IEEE Transactions on Medical Imaging*, vol. 18, no. 11, pp. 1085, 1999.

[10] J. Bobin, Y. Moudden, J.-L. Starck, and J. Fadili, Sparsity constraints for hyperspectral data analysis: linear mixture model and beyond, *In: Wavelets XIII. Proceedings of SPIE. No.7446.* Society of Photo-Optical Instrumentation Engineers (SPIE), Bellingham, WA, 2009.

ISBN 978-0-9928626-7-1 © EURASIP 2017