ROBUST DICTIONARY LEARNING USING α-DIVERGENCE

Asif Iqbal and Abd-Krim Seghouane

Department of Electrical and Electronic Engineering,
University of Melbourne, Australia

ABSTRACT

In this paper, a robust sequential dictionary learning (DL) algorithm is presented. It is obtained by using a robust loss function in the data fidelity term of the DL objective instead of the usual quadratic loss. The proposed robust loss function is derived from the α-divergence as an alternative to the Kullback-Leibler divergence which leads to a quadratic loss. Compared to other robust approaches, the proposed loss has the advantage of belonging to class of redescending M-estimators, guaranteeing inference stability for large deviation from the Gaussian nominal noise model. The algorithm is derived via adaptive sequential penalized rank-1 matrix approximation using a block coordinate descent approach to obtain the vector pairs of different rank-1 matrices. Performance comparison with similar robust DL algorithms on digit recognition highlights efficacy of the proposed algorithm.

Index Terms— Robust estimation, dictionary learning, α-divergence, outlier suppression

1. INTRODUCTION

Sparse signal representation [1] is enjoying a lot of attention from the research community and has been successfully applied to a range of signal processing applications. A signal admitting sparse representation can be represented by using only a few functions from a basis set (Fourier, DCT, wavelets). The choice of a basis set under which a given set of signals admits an efficient sparse approximation is crucial, and researchers [2] have shown that learning a basis (dictionary) from the data itself improves its sparse approximation.

In the past 1.5 decades, to take advantage of sparse approximations, a lot of dictionary learning (DL) methods have been proposed for a variety of signal processing applications. Among these one can cite image restoration [3], fMRI signal analysis [4–7], and face recognition [8] etc. Most popular dictionary learning (DL) methods [2][9] pose the DL problem with an ℓ_2-norm fidelity term and a sparse regularization term. These data-driven methods assume the Gaussian noise prior, leading to a squared ℓ_2-term as the maximum likelihood estimate. If such prior holds, the resulting dictionaries have been shown to achieve state-of-the-art performance. On the other hand, if the training data is contaminated i.e., contains anomalous observations (*outliers*), the ℓ_2-norm loss function, being sensitive to outliers, offers no protection against them.

The effects of such outliers can be mitigated by utilizing methods developed in robust statistics [10]. By making assumptions on outlier statistics, they develop learning methods that are less sensitive to their presence. Several robust DL methods have been developed recently using tools from area of robust statistics [11–14]. These methods replace the ℓ_2-norm data fidelity term with ℓ_1-norm error, also called the *least absolute deviation* [10, Ch. 7.11], Huber loss, or truncated ℓ_1-norm error term to counter the effects of outliers, leading to an outlier resistant robust dictionary estimate.

In this paper, we propose a novel robust DL algorithm that minimizes assumption on the noise. This is done by deriving a loss function from the α-divergence [15] and using it as the data fidelity term, instead of the quadratic loss widely used in the DL objective functions. The derived loss function belongs to the class of redescending M-estimators which guarantees stability of inference for deviations from the Gaussian nominal noise model [10]. The proposed algorithm is obtained via adaptive sequential penalized rank-1 matrix approximation, where a block coordinate descent approach is used to determine the vector pairs for the different rank-1 approximation matrices. The adaptive aspect of the algorithm allows different amount of shrinkage to be used for different entries of the sparse code matrix $\mathbf{X}$ [16].

2. BACKGROUND

Given a collection of signals $\mathbf{Y} \in \mathbb{R}^{n\times N}$, under the DL framework, an overcomplete dictionary $\mathbf{D} \in \mathbb{R}^{n\times K}$ and a sparse coefficient matrix $\mathbf{X} \in \mathbb{R}^{K\times N}$ can be obtained by optimizing the following objective function [16]

$$\min_{\mathbf{D}\in\mathcal{D},\mathbf{X}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 + \lambda\|\mathbf{X}\|_1 \qquad (1)$$

where the $\|\mathbf{X}\|_1 = \sum_{i=1}^{N}\sum_{j=1}^{K}|x_i^j|$, x_i^j is the entry at i^{th} column and j^{th} row of $\mathbf{X}$, λ is the sparsity regularization parameter, and $\mathcal{D} = \{\mathbf{d}_j \in \mathbb{R}^n \mid \|\mathbf{d}_j\|_2 = 1\ \forall j\}$. The objective in (1) is non-convex, however, it admits a multi-convex structure [17]. Thus, convergence to a local minima is possible us-

This work was supported by the Australian Research Council; grant FT 130101394.

978-1-5386-4658-8/18/$31.00 ©2019 IEEE

ing an alternating optimization strategy. Using this strategy, most DL algorithms solve the problem in two stages, first fixing $\mathbf{D}$, they perform ℓ_1-norm minimization for which various efficient methods exist [18].Second, with $\mathbf{X}$ fixed, (1) is minimized over $\mathbf{D}$. To do so, columns (atoms) of the dictionary $\mathbf{D}$ are either updated simultaneously [9][19] or sequentially [2][20], where the global minimization (1) (w.r.t. $\mathbf{D}$) is decomposed into K sub problems. DL algorithms alternate between these two stages until convergence.

An alternative DL approach has been adopted in [16, 21–23], where the authors aim to approximate the dataset $\mathbf{Y}$ by a penalized sum of rank-1 matrices, with the factors of such rank-1 matrices approximated using a simple and exact block coordinate descent approach. This approach works directly on the residual $\mathbf{E}_j = \mathbf{Y} - \sum_{i=1,i\neq j}^{K} \mathbf{d}_i \mathbf{x}^i$ to generate new pairs $(\mathbf{d}_j, \mathbf{x}^j)$ using the objective

$$\{\mathbf{d}_j, \mathbf{x}^j\} = \arg\min_{\mathbf{d}_j,\mathbf{x}^j} \|\mathbf{E}_j - \mathbf{d}_j\mathbf{x}^j\|_F^2 + \lambda\|\mathbf{x}^j\|_1 \text{ s.t. } \mathbf{D} \in \mathcal{D}. \quad (2)$$

The resulting algorithm can be seen as a penalized variant of alternating least squares [24] or power method for computing the SVD, where the ℓ_1-norm penalty is used to learn sparse $\mathbf{x}^j$. The estimates of $\mathbf{d}_j$ and $\mathbf{x}^j$ are then given by

$$\mathbf{d}_j = \frac{\mathbf{E}_j {\mathbf{x}^j}^\top}{||\mathbf{E}_j {\mathbf{x}^j}^\top||_2}, \quad (3)$$

$$\mathbf{x}^j = \text{sgn}(\mathbf{d}_j^\top \mathbf{E}_j) \circ \left(|\mathbf{d}_j^\top \mathbf{E}_j| - \frac{\lambda}{2} \mathbf{1}_{(N)}^\top \right)_+ \quad (4)$$

where $\circ$, $|\ .\ |$, $\text{sign}\,(.)$, $(x)_+$ define the entrywise variants of Hadamard product, absolute value, sign, and $\max(0, x)$ functions respectively. The $\mathbf{1}_N$ is a vector of ones of size N. As illustrated by different experimental results [16,21], this alternative DL scheme leads to improved performance compared to state-of-the-art methods. In the next section, we adopt this approach to develop an extension of [16] for robust DL.

3. PROPOSED DICTIONARY LEARNING APPROACH

To induce robustness into our dictionary estimates, we propose to use a loss function from the class of redescending M-estimators which guarantees stability of inference for deviations from the Gaussian nominal noise model [10]. The function is derived from the α-divergence [15], however, due to the lack of space, the derivation can not be included here. The loss function ℓ_α (termed as Gaussian fidelity) is defined as

$$\ell_\alpha(v) = \frac{1}{\alpha}\left(1 - \exp\left(\frac{-\alpha v^2}{2}\right)\right). \quad (5)$$

If $\alpha \to 0$, we have $\ell_\alpha(v) \to \ell_0(v) = v^2/2$, and the familiar quadratic loss associated with the Frobenius norm is recovered, which is highly sensitive to outliers in the data. The ℓ_2-norm, ℓ_1-norm and Gaussian fidelity terms are shown in Fig. 1 a). The case $\alpha > 0$, corresponds to a weighted estimation that tends to down weight the errors that are far from the nominal density, thus mitigating effects of the outliers.

The proposed robust DL algorithm is derived by using a variant of (1) where the function ℓ_α (5), shown above, is used instead of the Frobenuis norm in the fidelity term to give

$$\min_{\mathbf{D},\mathbf{X}} \ell_\alpha(\mathbf{Y} - \mathbf{DX}) + \lambda \sum_{i=1}^{N}\sum_{j=1}^{K} |\ x_i^j\ | \text{ s.t. } \mathbf{D} \in \mathcal{D} \quad (6)$$

where

$$\ell_\alpha(\mathbf{Y} - \mathbf{DX}) = \sum_{i=1}^{N}\sum_{m=1}^{n} \ell_\alpha(y_i^m - \mathbf{d}^m \mathbf{x}_i) \quad (7)$$

where ℓ_α is defined in (5), x_i^j is the entry at i^{th} column and j^{th} row of $\mathbf{X}$, $\mathbf{d}^m$ is the m^{th} row of $\mathbf{D}$, and $\mathbf{x}_i$ is the i^{th} column of $\mathbf{X}$. The proposed procedure is robust in the sense that DL is carried out using an objective function guaranteeing a stable learning in the presence of outliers. To solve the problem in (6), we start by expressing the matrix $\mathbf{DX}$ as a sum of K rank-1 matrices i.e. $\sum_{j=1}^{K} \mathbf{d}_j \mathbf{x}^j$, and propose an exact block coordinate descent approach to estimate the various rank-1 matrices as done in [16, 21]. Thus, we start by writing (6) as

$$\{\mathbf{d}_j, \mathbf{x}^j\} = \min_{\mathbf{d}_j \in \mathcal{D}, \mathbf{x}^j} \sum_{i=1}^{N} \ell_\alpha\left(\mathbf{e}_{ij} - \mathbf{d}_j x_i^j\right) + \lambda |x_i^j| \quad (8)$$

where $\mathbf{e}_{ij}$ is the i^{th} column of $\mathbf{E}_j$ and $\mathbf{E}_j = \mathbf{Y} - \sum_{i=1,i\neq j}^{K} \mathbf{d}_i \mathbf{x}^i$. The objective in (8) aims at estimating components of one rank-1 matrix at a time and to solve it, we propose an iterative alternating optimization strategy, i.e., during each iteration of the algorithm, we perform K penalized rank-1 matrix approximations. The updates for $\mathbf{d}_j$ and $\mathbf{x}^j$ are found by minimizing (8) one variable at a time, while fixing the other. Solutions for each of the variables are derived next.

3.1. Atom update

With a fixed $\mathbf{x}^j$, the update for $\mathbf{d}_j$ is found by solving

$$\mathbf{d}_j = \min_{\mathbf{d}_j \in \mathcal{D}} \sum_{i=1}^{N} \frac{1}{\alpha}\left(1 - \exp\left(-\frac{\alpha}{2}\left(\mathbf{e}_{ij} - \mathbf{d}_j x_i^j\right)^2\right)\right) \quad (9)$$

Differentiating (9) w.r.t. $\mathbf{d}_j$ and setting it to zero, we get

$$\sum_{i=1}^{N} \exp\left(-\frac{\alpha}{2}\left(\mathbf{e}_{ij} - \mathbf{d}_j x_i^j\right)^2\right)\left(\mathbf{d}_j {x_i^j}^2 - \mathbf{e}_{ij} x_i^j\right) = 0 \quad (10)$$

with the solution for $\mathbf{d}_j$ given by

$$\mathbf{d}_j = \left(\sum_{i=1}^{N} \mathbf{W}_i {x_i^j}^2\right)^{-1} \left(\sum_{i=1}^{N} \mathbf{W}_i \mathbf{e}_{ij} x_i^j\right) \quad (11)$$

followed by ℓ_2-normalization of $\mathbf{d}_j$. Here $\mathbf{W}_i \in \mathbb{R}^{n\times n}$ is a diagonal weighting matrix with $\exp(-\alpha(\mathbf{e}_{ij} - \mathbf{d}_j x_i^j)^2/2)$ as the weights of the diagonal.

3.2. Sparse code update

Next, we fix the atom $\mathbf{d}_j$, and find an update for $\mathbf{x}^j$ by solving

$$x_i^j = \min_{x_i^j} \ell_\alpha\left(\mathbf{e}_{ij} - \mathbf{d}_j x_i^j\right) + \lambda |x_i^j| \quad \forall\, i = 1, \ldots, N. \quad (12)$$

Taking derivative of (12) w.r.t. $\mathbf{x}_i^j$ and setting it to zero, we get

$$\frac{1}{\sigma}\mathbf{d}_j^\top \mathbf{W}_i \left(\mathbf{e}_{ij} - \mathbf{d}_j x_i^j\right) + \lambda \operatorname{sign}\left(x_i^j\right) = 0 \qquad (13)$$

here $\mathbf{W}_i$ is the weighting matrix and the solution for x_i^j is given by

$$\begin{aligned} x_i^j &= \operatorname{sign}(\eta_i)\left(|\eta_i| - \tilde{\lambda}_i\right)_+ \quad \forall\, i = 1, \ldots, N \\ \text{with } \eta_i &= \frac{\mathbf{d}_j^\top \mathbf{W}_i \mathbf{e}_{ij}}{\mathbf{d}_j^\top \mathbf{W}_i \mathbf{d}_j} \text{ and } \tilde{\lambda}_i = \frac{\lambda}{\mathbf{d}_j^\top \mathbf{W}_i \mathbf{d}_j}. \end{aligned} \qquad (14)$$

The resulting $(\mathbf{d}_j, \mathbf{x}^j)$ updates can be seen as a variant of the iterative re-weighted least squares method for rank-1 matrix approximation and based on our experience, only requires $3-6$ iterations of (11) and (14) to reach a local minima.

3.3. Weights & parameter selection

The weighting function resulting from differentiation of (8) with respect to either $\mathbf{d}_j$ or x_i^j, leading to a weighted least squares estimating equation is given by

$$\mathbf{W}_i\left(\mathbf{e}_{ij}, \mathbf{d}_j, x_i^j, \alpha\right) = \exp\left(-\alpha/2\left(\mathbf{e}_{ij} - \mathbf{d}_j x_i^j\right)^2\right) \quad (15)$$

with weights that are nearly zero for outliers (i.e., when the residual matrix entry $\mathbf{e}_{ij}$ is far from $\mathbf{d}_j x_i^j$.

The special case $\alpha = 0$ (used in first iteration of the algorithm), corresponds to uniform weights $w_1 = \cdots = w_n = 1$ and the implied learning algorithm is the one obtained by the Frobenious norm [16]. On the other hand, if $\alpha > 0$, entries of $\mathbf{e}_{ij}$ far from the $\mathbf{d}_j x_i^j$ receives relatively low weight compared to observations near $\mathbf{d}_j x_i^j$. Due to the form of the weights, anomalous observations far from bulk of the data are automatically down-weighted and have little impact on the final estimate making the learning algorithm robust against outliers.

The parameter α in (15) tunes the extent to which we down-weight anomalies in the data (see Fig. 1 c)), and its choice is of practical importance in applications. Let $\mathbf{e} \in \mathbb{R}^n$ be the residual vector and γ be the demarcation point such that if the square of an entry of $\mathbf{e}$ is above γ, its corresponding weight will be less than β. Thus to find the optimum parameter α, we need to find a relationship between α and γ. Such a relationship can be obtained from (15), and is given by $\alpha = -2\ln(\beta)/\gamma$, where β is the respective weight at demarcation point γ. In all experimental sections, we use $\beta = 0.5$ unless specified otherwise. The selection of γ is performed using the following procedure. Let $I = \lceil \delta n \rceil$, where $\delta \in (0,1)$ is a scalar, and $\lceil \delta n \rceil$ outputs smallest integer larger than δn. Then we set γ to be the I^{th} largest element from the set $\{e_m^2,\ m = 1, \ldots, n\}$. The parameter δ corresponds to the expected proportion of outliers present in $\mathbf{e}$.

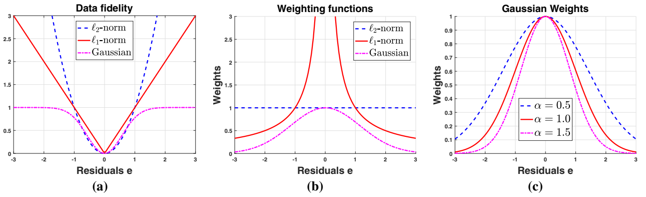

Fig. 1. (a) ℓ_2, ℓ_1, and Gaussian fidelity (5) terms, (b) weight functions, (c) Gaussian weights for different α values.

Algorithm 1: Proposed robust DL algorithm.

```
Input: Data matrix $\mathbf{Y} \in \mathbb{R}^{n\times N}$, $\mathbf{D} \in \mathbb{R}^{n\times K}$, $\lambda$, $\delta$,
       noIt
Initialization: Set $\mathbf{X} = 0$, $\epsilon = 10^{-4}$, and
    $\mathbf{W}_i = \mathbf{I}$, $\forall\, i = 1, \ldots, N$.
for $t = 1 : noIt$ do
    for $j = 1 : K$ do
        Compute $\mathbf{E}_j = \mathbf{Y} - \sum_{i=1, i\neq j}^{K} \mathbf{d}_i \mathbf{x}^i$
        if $t \neq 1$ then
            Using $\delta$, compute $\mathbf{W}_i\ \forall\, i = 1, \ldots, N$ as
              outlined in section 3.3.
        while $\|\mathbf{d}_j^t - \mathbf{d}_j^{t-1}\|_2 > \epsilon$ do
            Compute $\eta_i$ and $\lambda_i$ using (14)
            $x_i^j = \operatorname{sign}(\eta_i)\left(|\eta_i| - \tilde{\lambda}_i\right)_+ \quad \forall\, i = 1, \ldots, N$
            $\mathbf{d}_j = \left(\sum_{i=1}^{N} \mathbf{W}_i {x_i^j}^2\right)^{-1}\left(\sum_{i=1}^{N} \mathbf{W}_i \mathbf{e}_{ij} x_i^j\right)$
            $\mathbf{d}_j = \mathbf{d}_j / \|\mathbf{d}_j\|_2$
Output: $\mathbf{D}, \mathbf{X}$
```

4. EVALUATION ON DIGIT RECOGNITION

Digit recognition is an integral part of any document processing systems. These system require large amount of training data to learn from, however, the data could contain different types of contaminations (noise, occlusion / outliers). One way of tackling such problems are to find and discard the highly contaminated signals before the learning process. This technique, however, becomes prohibitive as the training size increases. In such conditions, a technique which is robust against such contaminations could prove beneficial.

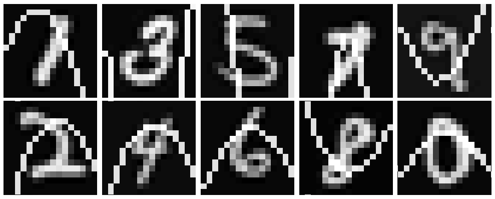

Fig. 2. Examples from MNIST training digits with outliers.

Table 1. Mean recognition results on contaminated test data.

	K	K-SVD	ORDL	L1-KSVD	Proposed
USPS	2	0.764	0.804	0.826	**0.844**
	3	0.775	0.819	0.845	**0.856**
	4	0.788	0.824	0.851	**0.870**
MNIST	2	0.677	0.696	0.719	**0.792**
	3	0.649	0.690	0.722	**0.797**
	4	0.649	0.669	0.726	**0.800**

Table 2. Mean recognition results on outlier free test datasets.

	K	K-SVD	ORDL	L1-KSVD	Proposed
USPS	2	0.805	0.824	0.852	**0.857**
	3	0.826	0.841	0.870	**0.883**
	4	0.837	0.854	0.879	**0.895**
MNIST	2	0.761	0.796	0.820	**0.845**
	3	0.760	0.813	0.849	**0.864**
	4	0.756	0.823	0.865	**0.878**

In this section we present performance comparison of the proposed robust DL method with respect to K-SVD [2], ORDL [13], and L1-KSVD [12]. We use USPS [25] and MNIST [26] handwritten digits datasets for the evaluation. Each dataset contains two sub-datasets for training and testing purposes. USPS dataset contains $7,291$ training and $2,007$ testing digit images, whereas, the MNIST dataset has $60,000$ training and $10,000$ testing digit images. We resized all gray-scale images to the same size of 16×16 pixels and normalized them w.r.t. the maximum pixel intensity. We generated outliers of the same size using a higher order polynomial with random parameters and added them to the training as well as testing images while making sure that the overlapping pixel intensities did not exceed 1. Few examples of corrupted images from MNIST training dataset are shown in Fig. 2. These corrupted training images were vectorized and placed as columns of a training matrix $\mathbf{Y} \in \mathbb{R}^{n\times N}$, where $n = 256$ and N represents the number of training samples.

We learn 10 class-specific dictionaries for each digit where the individual dictionaries were initialized with K samples from the respective training data itself. The sparsity parameters were set to $\lambda = 0.1$ for ORDL, L1-KSVD, and the proposed algorithm. For K-SVD, the sparsity parameter s was set to $s = K$. For the proposed algorithm, we set $\delta = 0.1$ and $\beta = 0.5$ for USPS and $\beta = 0.1$ for the MNIST datasets. All tuning parameters were selected using cross validation. The learning process was iterated over 40 iterations or was stopped early if $(\|\mathbf{D}_t - \mathbf{D}_{t-1}\|_F / \|\mathbf{D}_t\|_F < 0.01)$. The batch-size for ORDL was set to 250. Once the learning was complete, given a test sample $\mathbf{y}_i$, we used orthogonal matching pursuit (OMP) to solve

$$\mathbf{x}_i = \min_{\mathbf{z}} \|\mathbf{y}_i - \mathbf{D}\,\mathbf{z}\|_2^2 \ \text{ s.t. } \|\mathbf{z}\|_0 \leq 2K \tag{16}$$

where $\mathbf{D} \in \mathbb{R}^{n\times 10K}$ is the full dictionary. Using the resulting sparse vector $\mathbf{x}_i$, we calculate the representation error w.r.t. all class-specific dictionaries and select the one with smallest representation error. We repeated this procedure over $K = \{2, 3, 4\}$ and 20 trials, and the mean recognition results are reported in Table 1. We present the results on outlier free test datasets in Table 2 as well. From both tables, it is evident that the dictionaries learned by the proposed algorithm outperformed the competition. For visualization, the recovered dictionaries (for $K = 3$) are shown in Fig. 3. The figure shows that the proposed algorithm was able to fully reject the outliers while learning a set of clean representative atoms leading to a higher overall recognition accuracy.

5. CONCLUSION

In this paper we propose a robust dictionary learning algorithm using a loss function developed from α-divergence which guarantees stability of inference even for relatively large deviations from the Gaussian nominal noise model. The weighting function appearing during derivation of the solution down-weights high amplitude residuals, thus diminishing their effect on the dictionary estimate. Performance of the proposed algorithm is compared with state-of-the-art quadratic (ℓ_2) and ℓ_1-norm based DL methods on digit recognition application, highlighting its superior performance.

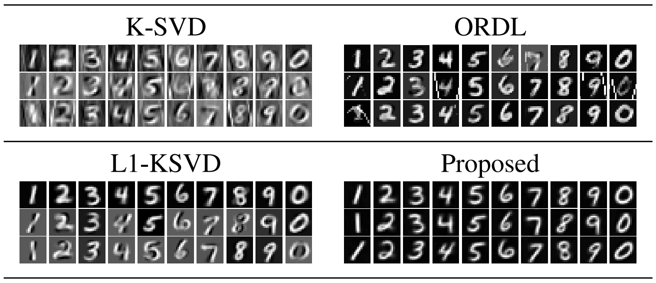

Fig. 3. Recovered dictionaries from MNIST dataset.

6. REFERENCES

[1] Z. Zhang et al., "A survey of sparse representation: Algorithms and applications," *IEEE Access*, vol. 3, pp. 490–530, 2015.

[2] M. Aharon et al., "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, pp. 4311–4322, 2006.

[3] A. K. Seghouane, A. Iqbal, and K. Abed Meraim, "A sequential block-structured dictionary learning algorithm for block sparse representations," *IEEE Transactions on Computational Imaging*, pp. 1–12, 2018.

[4] A. K. Seghouane and A. Iqbal, "Basis expansion approaches for regularized sequential dictionary learning algorithms with enforced sparsity for fMRI data analysis," *IEEE Transactions on Medical Imaging*, pp. 1796–1807, 2017.

[5] A. K. Seghouane and A. Iqbal, "Sequential dictionary learning from correlated data: Application to fMRI data analysis," *IEEE Transactions on Image Processing*, pp. 3002–3015, 2017.

[6] A. Iqbal et al., "Shared and subject-specific dictionary learning algorithm for multi-subject fMRI data analysis (ShSSDL)," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 11, pp. 2519–2528, Nov 2018.

[7] A. Iqbal and A. K. Seghouane, "A dictionary learning algorithm for multi-subject fMRI analysis based on a hybrid concatenation scheme," *Digital Signal Processing*, vol. 83, pp. 249–260, 2018.

[8] Y. Xu et al., "A survey of dictionary learning algorithms for face recognition," *IEEE Access*, vol. 5, pp. 8502–8514, 2017.

[9] K. Engan, S. O. Aase, and J. Hakon-Husoy, "Method of optimal directions for frame design," *IEEE Int. Conference on Acoustics, Speech, and Signal Processing*, pp. 2443–2446, 1999.

[10] P. J. Huber, *Robust Statistics*, Wiley series in probability and statistics. Wiley, New York, Hoboken, N.J., 2009.

[11] W. Jiang et al., "Robust dictionary learning with capped l1-norm," in *International Joint Conference on Artificial Intelligence*, 2015, pp. 3590–3596.

[12] S. Mukherjee et al., "L1-K-SVD: A robust dictionary learning algorithm with simultaneous update," *Signal Processing*, vol. 123, pp. 42–52, 2016.

[13] C. Lu et al., "Online robust dictionary learning," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 415–422.

[14] C. Zhao et al., "Background subtraction via robust dictionary learning," *EURASIP Journal on Image and Video Processing*, vol. 2011, no. 1, pp. 972961, 2011.

[15] A. Cichocki and S. I. Amari, "Families of alpha-Beta and Gamma-divergences: Flexible and robust measures of similarities," *Entropy*, vol. 12, pp. 1532–1568, 2010.

[16] A. K. Seghouane and A. Iqbal, "Consistent adaptive sequential dictionary learning," *Signal Processing*, vol. 99, pp. 2055–2065, 2018.

[17] J. Gorski et al., "Biconvex sets and optimization with biconvex functions: a survey and extensions," *Math. Meth. Oper. Res.*, vol. 66, pp. 373–407, 2007.

[18] J. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proceedings of the IEEE*, vol. 98, pp. 948–958, 2010.

[19] S. Ubaru, A. K. Seghouane, and Y. Saad, "Improving the incoherence of learned dictionary via rank shrinkage," *Neural Computation*, vol. 29, pp. 263–285, 2017.

[20] M. U. Khalid and A. K. Seghouane, "A single SVD sparse dictionary learning algorithm for fMRI data analysis," *In Proceedings of IEEE International Workshop on Statistical signal Processing*, pp. 65–68, 2014.

[21] A. K. Seghouane and M. Hanif, "A sequential dictionary learning algorithm with enforced sparsity," *IEEE International Conference on Acoustic Speech and signal Processing, ICASSP*, pp. 3876–3880, 2015.

[22] A. K. Seghouane and A. Iqbal, "A regularized sequential dictionary learning algorithm for fMRI data analysis," *In Proceedings of the IEEE Workshop on Machine Learning for Signal Processing, MLSP*, pp. 1–6, 2017.

[23] A. Iqbal and A. K. Seghouane, "An approach for sequential dictionary learning in nonuniform noise," *In Proceedings of the International Conference on Digital Image Computing: Techniques and Applications, DICTA*, pp. 1–6, 2017.

[24] G. H. Golub and C. f. Van Loan, *Matrix Computations*, Johns Hopkins, 1996.

[25] "Usps handwritten digits dataset," .

[26] Y. LeCun et al., "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.