

Through the Wall Radar Imaging via Kronecker-structured Huber-type RPCA

Hugo Brehier^{a,*}, Arnaud Breloy^b, Chengfang Ren^a, Guillaume Ginolhac^c

^a*SONDRA, CentraleSupélec, Gif-sur-Yvette, 91190, France*

^b*LEME, Université Paris-Nanterre, Ville d'Avray, 92410, France*

^c*LISTIC, Université Savoie Mont-Blanc, Annecy, 74940, France*

Abstract

The detection of multiple targets in an enclosed scene, from its outside, is a challenging topic of research addressed by Through-the-Wall Radar Imaging (TWRI). Traditionally, TWRI methods operate in two steps: first the removal of wall clutter then followed by the recovery of targets positions. Recent approaches manage in parallel the processing of the wall and targets via low rank plus sparse matrix decomposition and obtain better performances. In this paper, we reformulate this precisely via a RPCA-type problem, where the sparse vector appears in a Kronecker product. We extend this approach by adding a robust distance with flexible structure to handle heterogeneous noise and outliers, which may appear in TWRI measurements. The resolution is achieved via the Alternating Direction Method of Multipliers (ADMM) and variable splitting to decouple the constraints. The removal of the front wall is achieved via a closed-form proximal evaluation and the recovery of targets is possible via a tailored Majorization-Minimization (MM) step. The analysis and validation of our method is carried out using Finite-Difference Time-Domain (FDTD) simulated data, which show the advantage of our method in detection performance over complex scenarios.

Keywords: Through-the-Wall Radar Imaging, RPCA, Huber distance, Majorization-Minimization, ADMM, variable splitting

*Corresponding author

Email address: hugo.brehier@centralesupelec.fr (Hugo Brehier)

1. Introduction

Through the Wall Radar Imaging is a current topic of research (see e.g. [1] for a comprehensive review) that aims at detecting targets in an enclosed scene from its outside via radar measurements, the scene being unobservable to the naked eye. It makes use of the penetrative properties of electromagnetic waves to obtain returns from the inside of the scene while having to filter the front wall echoes. The ability to observe scenes through wall or other similar types of obstacle would be a useful technique for military operations, civilian rescue operations and monitoring [2, 3]. Many problems appear in the context of TWRI. Firstly, the front wall (facing the radar) returns are overwhelming and obscure the enclosed scene. Secondly, the echoes from the enclosed scene are subject to different phenomena: clutter from the inner walls gets mixed with the target echoes. Moreover, those returns can travel across different paths, so-called multipaths, which may create ghost targets.

Past works have focused on different aspects of the topic of TWRI such as localisation of targets, change detection, movement characterization [4, 5, 6, 7]. Here, we focus on the localisation of stationary targets which can be readily extended to moving targets by collecting measurement over time and applying the same methodology. We will focus on a 2D scenario which necessitates the use of multiple antennas (or a single travelling one) to achieve a sufficient resolution. A standard hypothesis in TWRI is for the wall to be homogeneous, with permittivity and thickness considered to be known, or to be estimated in a previous step [8, 9]. Other works have developed methods for the unknown case based on focusing techniques [10, 11]. In an earlier phase of TWRI, some methods [12, 13] were developed that use Synthetic Aperture Radar (SAR) techniques [14] such as Back-Projection (BP). Those methods require the acquisition of measurements from an empty scene to remove the front wall. Subsequently, two-step techniques were developed [15] which consist in: a) filtering the front wall echoes based on subspace decomposition [16, 17, 18] b) recovering the target positions, based on the hypothesis of sparsity of the targets w.r.t. the scene dimensions, with the possible use of Compressive Sensing methods to reduce computation times [19]. This approach requires the use of a dictionary to map the returns onto a grid covering the scene. This formalism also allows handling multipaths or front wall reflections more precisely [20]. Building on this, one-step methods have been explored during the past years via the framework of Robust Principal Component Analysis (RPCA) [21, 22, 23] which allows the joint

decomposition of a matrix in two separate components: one being low rank and the other being sparse, the two parts capturing respectively the returns of the front wall and the returns of the targets. Such one-step methods have been shown to perform better than their counterparts in several radar experiments [24, 25, 26, 27, 28].

We build upon these more recent approaches and address some of their limitations in the context of TWRI. A question to be raised is the robustness of those methods in the case that the measurements do not respect the model perfectly. Indeed, the returns from the front wall may not be homogeneous as supposed: the structure of drywall may for example create a discrepancy of the returns power among the different radar positions. Moreover, the permittivity of the wall that is supposed to be frequency-independent may be too restrictive and its treatment may lead to better performances. This has motivated us to inspect the addition of a robust distance [29] with flexible structure to one-step matrix decomposition methods applied to TWRI.

To do so, we first formalize the TWRI problem in the context of RPCA, which leads to the sparse component appearing in a Kronecker product, a special case of the model in [23]. We make adjustments to previous work in TWRI by using the ADMM framework [30]. This first part concluded with the presentation of a method, we then introduce the use of a robust distance with a flexible structure. This allows us to handle heterogeneous noise or outliers in the data closeness term, which may distort the results grossly with the usual euclidian distance. This was suggested in [31] but not developed. We present two methods for the resolution of this problem which make use of ADMM in tandem with variable splitting and either Proximal Gradient Descent (PGD) or MM frameworks.

The following sections of the paper are organized as follows. Section 2 presents a standard model for the measurements and describes existing two-step and one-step methods for TWRI. In Section 3, we introduce and develop the robust extension to one-step methods. Section 4 follows with experiments done on simulated data to compare the performance of the different methods. Finally, Section 5 summarizes the advantages of our method and perspectives of future work.

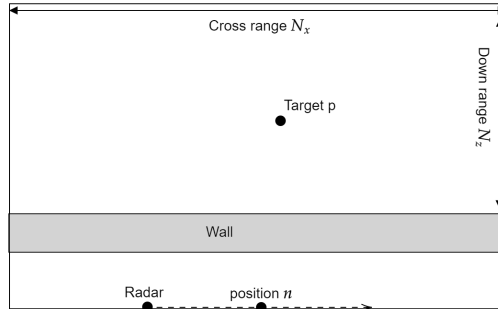


Figure 1: 2D Through the Wall setting (view from above)

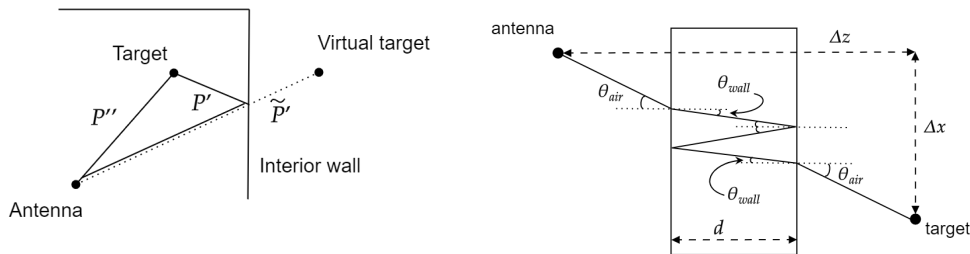


Figure 2: Multipath propagation via reflection at an interior wall (left) and via reverberation i.e. "wall ringing" (right)

2. Existing TWRI models and methods

2.1. Setting and signal model

We first present a signal model widely used in the TWRI literature [15, 25]. Consider a 2D scene, as in Figure 1, with homogeneous wall of thickness d and permittivity ϵ located along the x -axis. We obtain N measurements over a synthetic array parallel to the wall at a standoff distance z_{off} , with a stepped-frequency radar signal of M frequencies uniformly spaced over some frequency band $[\omega_1, \omega_M]$ so that: $\omega_m = \omega_1 + m\Delta\omega$, $m = 1, \dots, M$ with $\Delta\omega$ the frequency step. Measurements are in a stop-and-go setting.

The noiseless received signal can be written as the superposition of the returns from the front-wall and targets. This leaves out from the model echoes stemming from the inner walls which may form clutter in real-world

scenarios. For the m^{th} frequency and n^{th} position, we have:

$$y(m, n) = \sum_{k=1}^K \sigma_w^{(k)} \exp(-j\omega_m \tau_w^{(k)}) + \sum_{i=1}^R \sum_{p=1}^P \sigma_p^{(i)} \exp(-j\omega_m \tau_{p,n}^{(i)}) \quad (1)$$

where P is the number of point targets in the scene, considered to be low w.r.t. the scene dimensions, K is the number of reverberations in the front wall, while R is the number of possible multipaths, as the propagation through the front wall and the lateral/back walls induce several possible paths as presented in Figure 2.

Moreover, $\sigma_w^{(k)}$ is a complex-valued attenuation coefficient comprising the reflectivity of the wall and the path loss for the k^{th} wall reverberation, $\tau_w^{(k)}$ is the round trip propagation delay from transceiver to wall.

Additionally, $\sigma_p^{(i)}$ is a complex-valued attenuation coefficient which factors in the different losses for the i^{th} multipath to the p^{th} target: the wall refraction loss, path loss in air and wall, and target reflection loss. The two-way propagation delay from n^{th} transceiver to the p^{th} target along the i^{th} multipath is denoted $\tau_{p,n}^{(i)}$. The direct trajectory through the front wall can be computed by numerical methods as in [32], which allows us to evaluate the associated propagation delay.

For numerical evaluation and implementation, we discretize the scene into a grid of dimension $N_x \times N_z$ in crossrange vs downrange. We now denote $\tau_{n_x n_z, n}^{(i)}$ the propagation delay to the $(n_x, n_z)^{\text{th}}$ pixel for the i^{th} multipath scheme and the n^{th} transceiver position. Then, we can write the received signal through a dictionary Ψ which maps the whole scene. For the i^{th} multipath scheme and the n^{th} transceiver position, its $(n_x, n_z)^{\text{th}}$ column describes the return from a point target at the $(n_x, n_z)^{\text{th}}$ pixel:

$$[\Psi_n^{(i)}]_{n_x n_z} = [\exp(-j\omega_0 \tau_{n_x n_z, n}^{(i)}) \dots \exp(-j\omega_{M-1} \tau_{n_x n_z, n}^{(i)})]^T \quad (2)$$

Then, $\Psi_n \in \mathbb{C}^{M \times N_x N_z R}$ is the part of the overall dictionary mapping from received signal to target positions from the n^{th} transceiver position. This allows to write in vector form the signal received at the n^{th} position:

$$\mathbf{y}_n = \mathbf{l}_n + \underbrace{[\Psi_n^{(1)} \Psi_n^{(2)} \dots \Psi_n^{(R)}]}_{=\Psi_n} \underbrace{\begin{bmatrix} \mathbf{r}^{(1)} \\ \mathbf{r}^{(2)} \\ \vdots \\ \mathbf{r}^{(R)} \end{bmatrix}}_{=\mathbf{r}} \quad (3)$$

$$\implies \mathbf{y}_n = \mathbf{l}_n + \Psi_n \mathbf{r}$$

where $\mathbf{l}_n \in \mathbb{C}^M$ contains the returns of the front wall and $\mathbf{r}^{(i)} \in \mathbb{C}^{N_x N_z}$ is the scene vector associated to the i^{th} multipath propagation scheme containing the back-scattered signal complex amplitudes over the grid covering the scene.

In this setting, the returns of the front wall are overwhelming. It is needed to filter them out in order to achieve the detection of targets. The most convenient and common technique is to separate the target and wall subspaces to mitigate the contribution to the measurements of the wall. Indeed, the radar displacement axis being chosen to be parallel to the front wall induces an invariance of the front wall returns along the different measurement positions. Coupled with their higher power compared with the scene behind, this calls for the use of a subspace decomposition. The filtering of the front wall followed by the detection of targets can be stated as a two-step method which we detail in the next section.

2.2. SR-CS: vectorized overall model and two-step methods

The method of Amin and Ahmad [15], which we denote SR-CS (for Sparse Recovery - Compressed Sensing) considers a vectorized model where the total signal model is created by stacking the measurements at the N radar positions in a long composite vector.

Indeed, let $\mathbf{y} \triangleq [\mathbf{y}_1^T \dots \mathbf{y}_N^T]^T$ and $\Psi_A \triangleq [\Psi_1^T \dots \Psi_N^T]^T$ so that:

$$\mathbf{y} = [\mathbf{l}_1^T, \dots, \mathbf{l}_N^T]^T + \Psi_A \mathbf{r} \quad (4)$$

SR-CS assumes that the front wall returns have been suppressed (see e.g. [17, 18]) so that $\mathbf{l}_n = \mathbf{0} \forall n = 1, 2, \dots, N$ and $\mathbf{y} = \Psi_A \mathbf{r}$.

Assuming that the number of targets is relatively low w.r.t. the scene dimensions, the vector of amplitudes \mathbf{r} is sparse. The recovery of \mathbf{r} with a sparsity regularization consists in a renowned problem of sparse recovery, the most famous example being the LASSO regression [33], which uses a ℓ_1 norm regularization. The use of the $\ell_{2,1}$ norm, a regularization used in order to promote grouped sparsity across rows [34], has been developed for multipath exploitation in [20]. It is defined as the sum of the euclidian norm of the rows of a matrix. Indeed, in our case, rows represent one pixel viewed across different multipaths, we then want our method to promote activation of whole rows, as the underlying scene is the same across multipaths. This can filter out multipath ghosts which appear at some position in an unstructured way, i.e. not across all multipaths, on the contrary of true targets.

2.3. KRPCA: matricized overall model and one-step methods

2.3.1. Data Model

The approach in Section 2 is a sequential method in two steps : *a*) filter the front wall, *b*) recover the target positions. Recent works [25, 28] suggest that a parallel recovery of both components can improve performances.

This can be considered through a decomposition of the data matrix, more precisely low rank plus sparse decomposition methods. This was notably developed in the framework of Robust PCA (RPCA) [21, 22] whose goal is to retrieve a low-dimensional subspace in which lie the data points, except for some outliers which are accounted for in a sparse matrix. It makes use of the ℓ_1 and nuclear norms for convex relaxation, known to be the convex envelopes of the ℓ_0 ‘norm’ (the number of non-zeros entries) and rank of a (bounded) matrix [35]. In [23], it was extended to a setting with a compressing operator acting on the sparse component. However, we may observe that our model is a special case of the aforementioned method. Indeed, note that we can write an overall matricized model for the observations:

$$\underbrace{[\mathbf{y}_1 \dots \mathbf{y}_N]}_{=\mathbf{Y}} = \underbrace{[\mathbf{l}_1 \dots \mathbf{l}_N]}_{=\mathbf{L}} + \underbrace{[\Psi_1 \dots \Psi_N]}_{=\Psi} \underbrace{\begin{bmatrix} \mathbf{r} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{r} \end{bmatrix}}_{=\mathbf{I}_N \otimes \mathbf{r}} \quad (5)$$

$$\implies \mathbf{Y} = \mathbf{L} + \Psi (\mathbf{I}_N \otimes \mathbf{r})$$

with \otimes denoting the Kronecker product. $\mathbf{Y} \in \mathbb{C}^{M \times N}$ the data matrix, $\mathbf{L} \in \mathbb{C}^{M \times N}$ a low-rank matrix of front wall returns, $\Psi \in \mathbb{C}^{M \times N_x N_z R N}$ a dictionary mapping to the target returns and $\mathbf{S} \in \mathbb{C}^{N_x N_z R N \times N}$ the associated sparse matrix containing the scene vector.

2.3.2. Problem statement

Some works of low-rank plus sparse matrix decomposition exist in the context of TWRI [24, 25]. Additionally, the work of [28], denoted KRPCA (for Kronecker-structured RPCA), proposed the following formulation to refine the model of [23]:

$$\begin{aligned} \min_{\mathbf{L}, \mathbf{r}} \quad & \|\mathbf{L}\|_* + \lambda \|\text{vec}^{-1}(\mathbf{r})\|_{2,1} \\ \text{s.t.} \quad & \mathbf{Y} = \mathbf{L} + \Psi (\mathbf{I}_N \otimes \mathbf{r}) \end{aligned} \quad (6)$$

In the following, we define $\mathbf{R} = \text{vec}^{-1}(\mathbf{r})$ for ease of notation with $\text{vec}^{-1}(\mathbf{r}) \triangleq [\mathbf{r}^{(0)} \mathbf{r}^{(1)} \dots \mathbf{r}^{(R-1)}] \in \mathbb{C}^{N_x N_z \times R}$ so that $\text{vec}^{-1}(\text{vec}(\mathbf{R})) = \mathbf{R}$.

The resolution of KRPCA can be tackled via the Alternating Direction Method of Multipliers (ADMM) [30]. The Augmented Lagrangian associated to (6) is:

$$l(\mathbf{L}, \mathbf{r}, \mathbf{U}) = \|\mathbf{L}\|_* + \lambda \|\mathbf{R}\|_{2,1} + \Re \langle \mathbf{U}, \mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \mathbf{r}) \rangle + \frac{\mu}{2} \|\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \mathbf{r})\|_F^2 \quad (7)$$

where \mathbf{U} is the matrix of Lagrange multipliers, λ is the sparsity regularization parameter and μ is the augmented Lagrangian penalty parameter.

The resolution for the three variables can be summarized as:

- The subproblem for \mathbf{L} is obtained via the soft thresholding operator on singular values, that is the proximal (see e.g. [36] for a comprehensive review) of the nuclear norm (with threshold λ), denoted D_λ . It thus consists of the so-called ℓ_1 -norm proximal, the soft-thresholding operator S_λ , applied on the singular values of a matrix. Recall that S_λ is defined element by element as: $[S_\lambda(\mathbf{A})]_{i,j} = \text{sgn}(a_{ij}) (|a_{ij}| - \lambda)_+$ where $\text{sgn}(x) = x/|x|$ if $x \neq 0$, else $\text{sgn}(x) = 0$ for $x \in \mathbb{C}$ is the complex sign function and $(x)_+ = \max(x, 0)$ for $x \in \mathbb{R}$. Also recall the Singular Value Decomposition (SVD) denoted by $\mathbf{A} \stackrel{\text{SVD}}{=} \mathbf{U} \Sigma \mathbf{V}^H$, so that:

$$D_\lambda(\mathbf{A}) = \mathbf{U} S_\lambda(\Sigma) \mathbf{V}^H \quad (8)$$

- The subproblem for \mathbf{r} is not solvable via a similar proximal evaluation. However, we may use proximal gradient descent (PGD) [36] since the objective function of this step is a sum of two convex terms with one being non-smooth. We can use a fixed step-size which is readily computed via the Hessian of the derivable part of the objective function. The proximal operator of $\ell_{2,1}$ -norm (with threshold λ), denoted T_λ , operating row by row over \mathbf{A} , is defined for the i^{th} row $\mathbf{A}_{i:}$ as:

$$[T_\lambda(\mathbf{A})]_{i:} = \left(1 - \frac{\lambda}{\|\mathbf{A}_{i:}\|} \right)_+ \mathbf{A}_{i:} \quad (9)$$

In fact, it is the proximal of the classical ℓ_2 -norm applied on a given row, as proximals are separable over sums.

- Finally, the subproblem for \mathbf{U} is a standard ADMM step of dual ascent. The interesting point is that the step-size is already known: it is the parameter of the Augmented Lagrangian.

The method is summarized in Algorithm 1.

Algorithm 1 Algorithm for KRPCA

- 1: Have: $\{\mathbf{y}_i\}_{i=1}^N, \{\Psi_i\}_{i=1}^N$
 - 2: Choose: λ, μ
 - 3: $\mathbf{Y} \triangleq [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N]$
 - 4: $\Psi \triangleq [\Psi_1, \Psi_2, \dots, \Psi_N]$
 - 5: $\Psi_A \triangleq [\Psi_1^T \Psi_2^T \dots \Psi_N^T]^T$
 - 6: $\mathbf{P} = \Psi_A^H \Psi_A$
 - 7: $t = 1/\lambda_{\max}(\mu\mathbf{P})$
 - 8: Initialize: $\mathbf{L}, \mathbf{R}, \mathbf{U}$
 - 9: **repeat**
 - 10: $\mathbf{L} = D_{1/\mu}(\mathbf{Y} - \Psi(\mathbf{I}_N \otimes \mathbf{r}) + \mu^{-1}\mathbf{U})$
 - 11: $\mathbf{\Gamma} = \mathbf{L} - \mathbf{Y} - \mu^{-1}\mathbf{U}$
 - 12: $\mathbf{n} = \Psi_A^H \text{vec } \mathbf{\Gamma}$
 - 13: **repeat:**
 - 14: $\mathbf{R} = T_{\lambda t}(\text{vec}^{-1}(\mathbf{r} - t\mu(\mathbf{n} + \mathbf{P}\mathbf{r})))$
 - 15: $\mathbf{r} = \text{vec}(\mathbf{R})$
 - 16: **until** stopping criterion is met
 - 17: $\mathbf{U} = \mathbf{U} + \mu(\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \mathbf{r}))$
 - 18: **until** stopping criterion is met
-

2.3.3. Convergence analysis

The convergence of the algorithm is assured by the theory surrounding ADMM. In our case, both functions in the objective functions are proper closed convex functions. Assuming the (non-augmented) Lagrangian has a saddle-point, this ADMM algorithm for KRPCA guarantees residual convergence, objective convergence and dual convergence [30].

2.3.4. Computational complexity

The computational complexity of the derived algorithm for KRPCA is dependant on some assumptions on the order of the dimensions considered. Assume that $MN > D > M > N$ where $D = N_x N_z R$ is the discretized scene

grid size for all multipaths (and recall that M, N are respectively the number of frequencies and radar snapshots). Under those assumptions, the major cost of the overall algorithm is the computation of \mathbf{P} during the initialization, which is of complexity $\mathcal{O}(MND^2)$. In the case of repeated calls of KRPCA (e.g. for Monte Carlo simulations), we can look only at the cost of the inner loop, considering \mathbf{P} as cached. Denoting that $\Psi(\mathbf{I}_N \otimes \mathbf{r}) = \text{vec}^{-1}(\Psi_A \mathbf{r})$, this operation can be seen to be of complexity $\mathcal{O}(MND)$. The PGD step has complexity $\mathcal{O}(KD^2)$. We assume that $K = 1$ or relatively small, which is respectable in practice, otherwise the ordering of the different dimensions becomes too tight to make general statements. We then conclude that the inner loop has complexity $\mathcal{O}(MND)$.

3. HKRPCA : handling outliers via a robust distance in low-rank plus sparse decomposition methods

The performance of KRPCA and other methods using a least squares data closeness term is susceptible to heterogeneous noise or outliers that may well appear in the context of TWRI. Indeed, as described in [37], most radar clutter types can be described as heterogeneous. For example, in the context of TWRI, a drywall will not have homogeneous returns in power across measurement positions. Moreover, the wall characteristics (permittivity and conductivity) may be dependant on frequency i.e. the wall is dispersive [1].

3.1. Problem statement

In order to alleviate the potential problems in estimation caused by heterogeneous noise or outliers, we set out to include a robust distance [38] in our problem formulation to model the data closeness.

This leads us to define the following optimization problem, which we call HKRPCA (for Huber-type KRPCA):

$$\min_{\mathbf{L}, \mathbf{R}} \|\mathbf{L}\|_* + \lambda \|\mathbf{R}\|_{2,1} + \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} H_c(\|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i}\|_F) \quad (10)$$

with \mathcal{P} a **partition** of the entries of the residual matrix with i^{th} element p_i and H_c the renowned Huber loss function Huber [29] with threshold $c \in \mathbb{R}^+$, defined $\forall x \in \mathbb{R}$ as:

$$H_c(x) = \begin{cases} \frac{1}{2}x^2 & \text{if } |x| \leq c \\ c(|x| - \frac{1}{2}c) & \text{if } |x| > c \end{cases} \quad (11)$$

The rationale behind such a function is that outliers are higher contributors to the data closeness term than other points. Having a linear term in the loss specifically for them will lower their influence while the inliers will contribute to the loss via a quadratic term, similarly to classical least squares.

The flexible block-wise partition of entries allows us to model the outliers shape as we see fit. For example, if the wall materials are structured rather than homogeneous, the noise power may be variable by radar position, which induces a column-wise heterogeneity that can be taken into account in a column-wise partition.

3.2. ADMM algorithm with a semi-split of variables

Handling the problem (10) directly can be achieved by proximal gradient descent alternated on the two variables. However, a strategy to obtain closed form updates is to introduce auxiliary variables to decouple the terms of the objective function. We introduce one auxiliary variable $\mathbf{M} = \mathbf{L}$ to decouple the nuclear norm from the Huber cost. We will see later that the split of \mathbf{r} does not yield a similar proximal closed form. We consider the problem:

$$\begin{aligned} \min_{\mathbf{L}, \mathbf{R}, \mathbf{M}} \quad & \|\mathbf{M}\|_* + \lambda \|\mathbf{R}\|_{2,1} + \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} H_c(\|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i}\|_F) \\ \text{s.t.} \quad & \mathbf{M} = \mathbf{L} \end{aligned} \tag{12}$$

This semi-splitting problem (12) can be tackled through the ADMM framework. The Augmented Lagrangian associated with (12) is:

$$\begin{aligned} l(\mathbf{L}, \mathbf{R}, \mathbf{M}, \mathbf{U}) = & \|\mathbf{M}\|_* + \lambda \|\mathbf{R}\|_{2,1} + \Re \langle \mathbf{U}, \mathbf{M} - \mathbf{L} \rangle + \frac{\nu}{2} \|\mathbf{M} - \mathbf{L}\|_F^2 \\ & + \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} H_c(\|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i}\|_F) \end{aligned} \tag{13}$$

As for KRPCA, the following subsections will detail the update of each variables for minimizing $l(\mathbf{L}, \mathbf{R}, \mathbf{M}, \mathbf{U})$.

3.2.1. L-update

For this variable, the minimization consists in finding:

$$\arg \min_{\mathbf{L}} \quad \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} H_c(\|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i}\|_F) + \frac{\nu}{2} \left\| \mathbf{M} - \mathbf{L} + \frac{1}{\nu} \mathbf{U} \right\|_F^2 \tag{14}$$

The resulting update solving for (14) is given in the following proposition.

Proposition 1. *The solution is $\forall p_i \in \mathcal{P}$:*

$$[\mathbf{L}]_{p_i} = \text{prox}_{(\mu/2\nu)H_c \circ \|\cdot\|_F} \left([\mathbf{M} + \frac{1}{\nu}\mathbf{U} - \mathbf{Y} + \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i} \right) + [\mathbf{Y} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i} \quad (15)$$

with the proximal defined in the proof below (equations (18) and (19)).

Proof. The problem (14) is separable in the blocks $\{[\mathbf{L}]_{p_i}\}$:

$$\begin{aligned} \min_{\{[\mathbf{L}]_{p_i}\}} & \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} H_c(\|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i}\|_F) \\ & + \frac{\nu}{2} \sum_{p_i \in \mathcal{P}} \left\| [\mathbf{M} - \mathbf{L} + \frac{1}{\nu}\mathbf{U}]_{p_i} \right\|_F^2 \end{aligned} \quad (16)$$

By the separability property of proximals [36], we can consider the proximal over each block separately:

$$\min_{[\mathbf{L}]_{p_i}} \frac{\mu}{2} H_c(\|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i}\|_F) + \frac{\nu}{2} \left\| [\mathbf{M} - \mathbf{L} + \frac{1}{\nu}\mathbf{U}]_{p_i} \right\|_F^2 \quad (17)$$

We then compute the proximal of $f(\mathbf{X}) = H_c(\|\mathbf{X} + \mathbf{B}\|_F)$ with \mathbf{B} a constant term. The proximal of the Huber function has a known form [39]:

$$\text{prox}_{aH_c}(x) = \left(1 - \frac{a}{\max(|\frac{x}{c}|, a+1)} \right) x \quad (18)$$

We can then leverage a theorem of norm composition [39] to get:

$$\text{prox}_{aH_c \circ \|\cdot\|_F}(\mathbf{X}) = \begin{cases} \text{prox}_{aH_c}(\|\mathbf{X}\|_F) \cdot \frac{\mathbf{X}}{\|\mathbf{X}\|_F} & \text{if } \mathbf{X} \neq \mathbf{0} \\ \mathbf{0} & \text{if } \mathbf{X} = \mathbf{0} \end{cases} \quad (19)$$

We finally use the translation properties of proximal operators, so that, $\forall p_i \in \mathcal{P}$, the update is:

$$[\mathbf{L}]_{p_i} = \text{prox}_{(\mu/2\nu)H_c \circ \|\cdot\|_F} \left([\mathbf{M} + \frac{1}{\nu}\mathbf{U} - \mathbf{Y} + \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i} \right) + [\mathbf{Y} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i} \quad (20)$$

□

This gives a closed-form update for the \mathbf{L} -step. We will see later that the auxiliary variable \mathbf{M} as well as the dual variable \mathbf{U} also have closed-forms.

3.2.2. \mathbf{R} -update: via PGD

The minimization problem over \mathbf{R} is:

$$\min_{\mathbf{R}} \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} H_c(\|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i}\|_F) + \lambda \|\mathbf{R}\|_{2,1} \quad (21)$$

It is possible to use proximal gradient descent (PGD) for the minimization over this variable. We will consider the vectorized variable \mathbf{r} to compute the gradient and unvectorize the solution to apply the proximal. At iteration $t + 1$, with step-size s , we have :

$$\mathbf{R}_{t+1} = T_{\lambda s} \left(\text{vec}^{-1} \left(\mathbf{r}_t - s \frac{\mu}{2} \mathbf{g}_t \right) \right) \quad (22)$$

where T is the row thresholding operator i.e. the proximal of the $\ell_{2,1}$ norm (see Equation (9)) and \mathbf{g} is the needed gradient of the sum of Huber functions.

Proposition 2. *The gradient \mathbf{g} w.r.t. \mathbf{r} is:*

$$\mathbf{g} = - \sum_{p_i \in \mathcal{P}} \frac{H'_c(\|[\mathbf{E}]_{p_i}\|_F)}{\|[\mathbf{E}]_{p_i}\|_F} \left(\sum_{(j,k) \in p_i} [\mathbf{E}]_{j,k} (\Psi_k)_{j,:}^H \right) \quad (23)$$

where $\mathbf{E} = \mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \mathbf{r})$ and $(\Psi_k)_{j,:}$ denotes the j^{th} line of Ψ_k .

Proof. The gradient is computed accordingly to Wirtinger calculus, since we have an objective function of complex variables. Gradient descent in this setting is achieved with:

$$\mathbf{g} = 2 \frac{d}{d\mathbf{r}^*} \left(\sum_{p_i \in \mathcal{P}} H_c(\|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \mathbf{r})]_{p_i}\|_F) \right) \quad (24)$$

Using the chain rule, we get:

$$\mathbf{g} = 2 \frac{d}{d\mathbf{r}^*} \sum_{p_i \in \mathcal{P}} H_c(\|[\mathbf{E}]_{p_i}\|_F) = \sum_{p_i \in \mathcal{P}} \frac{H'_c(\|[\mathbf{E}]_{p_i}\|_F)}{\|[\mathbf{E}]_{p_i}\|_F} \cdot \frac{d}{d\mathbf{r}^*} \|[\mathbf{E}]_{p_i}\|_F^2 \quad (25)$$

with the derivative of H_c being:

$$H'_c(x) = \begin{cases} x & \text{if } |x| \leq c \\ c \text{sgn}(x) & \text{if } |x| > c \end{cases} \quad (26)$$

where sgn denotes the sign function. Finally, we compute:

$$\begin{aligned} \frac{d}{d\mathbf{r}^*} \|\mathbf{[E]}_{p_i}\|_F^2 &= \sum_{(j,k) \in p_i} \frac{d}{d\mathbf{r}^*} |[\mathbf{Y}]_{j,k} - [\mathbf{L}]_{j,k} - (\Psi_k)_{j,:} \mathbf{r}|^2 \\ &= \sum_{(j,k) \in p_i} -([\mathbf{Y}]_{j,k} - [\mathbf{L}]_{j,k} - (\Psi_k)_{j,:} \mathbf{r})(\Psi_k)_{j,:}^H \end{aligned} \quad (27)$$

where $(\Psi_k)_{j,:} \mathbf{r}$ is a scalar as $(\Psi_k)_{j,:}$ denotes the j^{th} line of Ψ_k . Then:

$$\mathbf{g} = - \sum_{p_i \in \mathcal{P}} \frac{H'_c(\|\mathbf{[E]}_{p_i}\|_F)}{\|\mathbf{[E]}_{p_i}\|_F} \left(\sum_{(j,k) \in p_i} [\mathbf{E}]_{j,k} (\Psi_k)_{j,:}^H \right) \quad (28)$$

□

The step-size can be found by backtracking line-search (via Armijo's rule) which consists in iteratively shrinking an initially large step-size until sufficient decrease has been achieved. In practice, the step-size does not vary over iterations so that it can be fixed to one precomputed value (linked to the Lipschitz constant of the gradient above).

The gradient \mathbf{g} may be compactly written for faster implementation:

$$\mathbf{g} = -\Psi_g \text{bdiag}(\mathbf{e}_g) \mathbf{h}_g = -\Psi_g (\mathbf{e}_g \odot (\mathbf{h}_g \otimes \mathbf{1})) \quad (29)$$

where $\mathbf{1}$ is a vector of ones and \odot denotes the Hadamard product. The operator bdiag assigns a block diagonal matrix to a composite vector, Ψ_g collects the dictionary vectors in the innermost sum, \mathbf{e}_g the associated residues, and \mathbf{h}_g the fraction of norms in the outermost sum. Note that $\mathbf{e}_g \odot (\mathbf{h}_g \otimes \mathbf{1})$ is faster to compute than $\text{bdiag}(\mathbf{e}_g) \mathbf{h}_g$ as it avoids summing over the zeros of the block-diagonal matrix.

3.2.3. \mathbf{M} -update

Thanks to the variable split, the update \mathbf{M} appears as a classical proximal problem with closed form solution. Indeed, after completing the squared norm, the problem of solving (13) over \mathbf{M} consists in finding:

$$\arg \min_{\mathbf{M}} \|\mathbf{M}\|_* + \frac{\nu}{2} \left\| \mathbf{M} - \mathbf{L} + \frac{1}{\nu} \mathbf{U} \right\|_F^2 \quad (30)$$

which is a proximal of the nuclear norm. Thus :

$$\mathbf{M} = D_{1/\nu}(\mathbf{L} - \frac{1}{\nu} \mathbf{U}) \quad (31)$$

where D is the singular value thresholding operator (see Equation (8)).

3.2.4. U-update

Finally, the \mathbf{U} update is a standard step of ADMM, the dual ascent step:

$$\mathbf{U} = \mathbf{U} + \nu(\mathbf{M} - \mathbf{L}) \quad (32)$$

The method is summarized in Algorithm 2.

Algorithm 2 Algorithm for HKRPCA (semi variable splitting)

- 1: Have: $\{\mathbf{y}_i\}_{i=1}^N, \{\Psi_i\}_{i=1}^N$
 - 2: Choose: $\lambda, \mu, \nu, \eta, c, t$ and \mathcal{P}
 - 3: $\mathbf{Y} \triangleq [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N]$
 - 4: $\Psi \triangleq [\Psi_1, \Psi_2, \dots, \Psi_N]$
 - 5: $\Psi_A \triangleq [\Psi_1^T \Psi_2^T \dots \Psi_N^T]^T$
 - 6: Initialize: $\mathbf{L}, \mathbf{R}, \mathbf{M}, \mathbf{U}$
 - 7: **repeat**:
 - 8: $[\mathbf{L}]_{p_i} = \text{prox}_{(\mu/2\nu)H_c \circ \|\cdot\|_F} \left([\mathbf{M} + \frac{1}{\nu}\mathbf{U} - \mathbf{Y} + \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i} + [\mathbf{Y} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i} \quad \forall p_i \in \mathcal{P} \right)$
 - 9: **repeat**:
 - 10: $\mathbf{E} = \mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))$
 - 11: $\mathbf{G} = -\text{vec}^{-1} \left(\sum_{p_i \in \mathcal{P}} \frac{H'_c(\|\mathbf{E}\|_{p_i})}{\|\mathbf{E}\|_{p_i}} \left(\sum_{(j,k) \in p_i} [\mathbf{E}]_{j,k} (\Psi_k)^H_{j,:} \right) \right)$
 - 12: $\mathbf{R} = T_{\lambda s} (\mathbf{R} - s \frac{\mu}{2} \mathbf{G})$
 - 13: **until** stopping criterion is met
 - 14: $\mathbf{M} = D_{1/\nu} (\mathbf{L} - \frac{1}{\nu} \mathbf{U})$
 - 15: $\mathbf{U} = \mathbf{U} + \nu(\mathbf{M} - \mathbf{L})$
 - 16: **until** stopping criterion is met
-

3.3. ADMM algorithm with full variable splitting

The update for \mathbf{r} via PGD is not the only option, we may avoid the use of an unknown step-size tuned via linesearch by splitting the variable similarly to \mathbf{L} to decouple the terms in appears in. If we take this route, the formulation is:

$$\begin{aligned} \min_{\mathbf{L}, \mathbf{R}, \mathbf{M}, \mathbf{S}} \quad & \|\mathbf{M}\|_* + \lambda \|\mathbf{S}\|_{2,1} + \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} H_c(\|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i}\|_F) \\ \text{s.t.} \quad & \mathbf{M} = \mathbf{L}, \quad \mathbf{S} = \mathbf{R} \end{aligned} \quad (33)$$

This full variable splitting problem (33) can be tackled through the ADMM framework. The Augmented Lagrangian associated with (33) is:

$$\begin{aligned}
l(\mathbf{L}, \mathbf{R}, \mathbf{M}, \mathbf{S}, \mathbf{U}, \mathbf{V}) &= \|\mathbf{M}\|_* + \lambda \|\mathbf{S}\|_{2,1} \Re \langle \mathbf{U}, \mathbf{M} - \mathbf{L} \rangle + \frac{\nu}{2} \|\mathbf{M} - \mathbf{L}\|_F^2 \\
&+ \Re \langle \mathbf{V}, \mathbf{S} - \mathbf{R} \rangle + \frac{\eta}{2} \|\mathbf{S} - \mathbf{R}\|_F^2 \\
&+ \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} H_c(\|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i}\|_F)
\end{aligned} \tag{34}$$

3.3.1. $\mathbf{L}, \mathbf{M}, \mathbf{U}$ -updates

The \mathbf{L} , \mathbf{M} and \mathbf{U} updates do not change from the semi variable splitting method. Indeed, the major difference is in the \mathbf{r} update.

3.3.2. \mathbf{R} -update via MM

The objective function is in this case:

$$\min_{\mathbf{R}} \quad \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} H_c(\|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i}\|_F) + \frac{\eta}{2} \left\| \mathbf{S} - \mathbf{R} + \frac{1}{\eta} \mathbf{V} \right\|_F^2 \tag{35}$$

Via decoupling, we cannot find a similar closed-form proximal evaluation for \mathbf{r} as for \mathbf{L} in Proposition 1. Indeed, the sum of Huber functions is not separable over \mathbf{r} . Instead, we will show that the Majorization-Minimization (MM) framework [40] gives us a way to solve for this subproblem iteratively. The MM framework consists in finding a local majorizing surrogate, minimizing it and iterating those steps.

Proposition 3. *A MM scheme can be tailored which converges to a critical point of (35), with iteration $t + 1$:*

$$\begin{aligned}
\mathbf{r}_{t+1} &= \left(\frac{\mu}{2} \Psi_{AW(\mathbf{r}_t)}^H \Psi_{AW(\mathbf{r}_t)} + \eta \mathbf{I} \right)^{-1} \times \\
&\quad \left(\frac{\mu}{2} \Psi_{AW(\mathbf{r}_t)}^H (\text{vec } \mathbf{Y}_{W(\mathbf{r}_t)} - \text{vec } \mathbf{L}_{W(\mathbf{r}_t)}) + (\eta \text{vec } \mathbf{S} + \text{vec } \mathbf{V}) \right)
\end{aligned} \tag{36}$$

with \mathbf{W} depending on \mathbf{r}_t , which we drop from notations below. We have $\mathbf{L}_W = \mathbf{W} \odot \mathbf{L}$, $\mathbf{Y}_W = \mathbf{W} \odot \mathbf{Y}$, $\Psi_{AW} = \text{vec}(\mathbf{W}) \mathbf{1}^T \odot \Psi_A$ and \mathbf{W} is defined by $[\mathbf{W}]_{j,k} = w_i(\mathbf{r}_t)$ where the $(j, k)^{\text{th}}$ entry is in the i^{th} patch, with $w_i^2(\mathbf{r}_t) = 1$ if $e_i(\mathbf{r}_t) \leq c$ or else $w_i^2(\mathbf{r}_t) = \frac{c}{e_i(\mathbf{r}_t)}$ where $e_i(\mathbf{r}_t) = \|[\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \mathbf{r}_t)]_{p_i}\|_F$

Proof. Consider the vectorized variable \mathbf{r} whose update we can unvectorize for \mathbf{R} . The first step is to find a majorizing function of $H_c(x)$ at some point x_t that we will denote $G_c(x|x_t)$. It must be equal to H_c at the point x_t and greater at all other points. We can use the result from [41, Theorem 4.5] :

$$G_c(x|x_t) = \frac{H'_c(x_t)}{2x_t}(x^2 - x_t^2) + H_c(x_t) \quad (37)$$

This is the sharpest quadratic majorizer. We can obtain:

$$G_c(x|x_t) = \begin{cases} \frac{1}{2}x^2 & \text{if } |x_t| \leq c \\ \frac{1}{2}\frac{c}{|x_t|}x^2 + \frac{1}{2}c(|x_t| - c) & \text{if } |x_t| > c \end{cases} \quad (38)$$

Note $\forall p_i \in \mathcal{P}$ that $e_i(\mathbf{r}) = \|\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \mathbf{r})\|_{p_i}$ and $e_i(\mathbf{r}_t)$ is the same quantity but with \mathbf{r}_t , the variable at the previous MM iteration.

By the definition of G just above, we can write:

$$\arg \min_{\mathbf{r}} G_c(e_i(\mathbf{r})|e_i(\mathbf{r}_t)) = \arg \min_{\mathbf{r}} \frac{1}{2}w_i^2(\mathbf{r}_t)e_i^2(\mathbf{r}) \quad (39)$$

where $w_i^2(\mathbf{r}_t) = 1$ if $e_i(\mathbf{r}_t) \leq c$ or else $w_i^2(\mathbf{r}_t) = \frac{c}{e_i(\mathbf{r}_t)}$. Also note that we can sum the majorizers over all blocks to get a global one. Then, it follows that by adding the remaining quadratic term of the objective function, we get the following majorizer at point \mathbf{r}_t to the objective function (35) :

$$\mathcal{G}_c(\mathbf{r}|\mathbf{r}_t) = \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} G_c(e_i(\mathbf{r})|e_i(\mathbf{r}_t)) + \frac{\eta}{2} \left\| \mathbf{r} - (\text{vec } \mathbf{S} + \frac{1}{\eta} \text{vec } \mathbf{V}) \right\|_F^2 \quad (40)$$

So that, via the MM framework, we are left with finding:

$$\begin{aligned} \mathbf{r}_{t+1} &= \arg \min_{\mathbf{r}} \mathcal{G}_c(\mathbf{r}|\mathbf{r}_t) \\ &= \arg \min_{\mathbf{r}} \frac{\mu}{4} \sum_{p_i \in \mathcal{P}} w_i^2(\mathbf{r}_t)e_i^2(\mathbf{r}) + \frac{\eta}{2} \left\| \mathbf{r} - (\text{vec } \mathbf{S} + \frac{1}{\eta} \text{vec } \mathbf{V}) \right\|_F^2 \\ &= \arg \min_{\mathbf{r}} \frac{\mu}{4} \|\mathbf{W} \odot (\mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \mathbf{r}))\|_F^2 \\ &\quad + \frac{\eta}{2} \left\| \mathbf{r} - (\text{vec } \mathbf{S} + \frac{1}{\eta} \text{vec } \mathbf{V}) \right\|_F^2 \end{aligned} \quad (41)$$

where \mathbf{W} is such that $[\mathbf{W}]_{j,k} = w_i(\mathbf{r}_t)$ where the $(j, k)^{th}$ entry is in the i^{th} patch. To find the minimizer in (41), we vectorize the first term since the Frobenius norm acts component-wise. Then:

$$\begin{aligned} \mathbf{r}_{t+1} = \arg \min_{\mathbf{r}} & \quad \frac{\mu}{4} \|\Psi_{AW} \mathbf{r} - (\text{vec } \mathbf{Y}_W - \text{vec } \mathbf{L}_W)\|_F^2 \\ & + \frac{\eta}{2} \left\| \mathbf{r} - \left(\text{vec } \mathbf{S} + \frac{1}{\eta} \text{vec } \mathbf{V} \right) \right\|_F^2 \end{aligned} \quad (42)$$

where $\mathbf{L}_W = \mathbf{W} \odot \mathbf{L}$, $\mathbf{Y}_W = \mathbf{W} \odot \mathbf{Y}$ and $\Psi_{AW} = \text{vec}(\mathbf{W}) \mathbf{1}^T \odot \Psi_A$. Via the first order optimality conditions, we get:

$$\begin{aligned} \mathbf{r}_{t+1} = & \left(\frac{\mu}{2} \Psi_{AW(\mathbf{r}_t)}^H \Psi_{AW(\mathbf{r}_t)} + \eta \mathbf{I} \right)^{-1} \times \\ & \left(\frac{\mu}{2} \Psi_{AW(\mathbf{r}_t)}^H (\text{vec } \mathbf{Y}_{W(\mathbf{r}_t)} - \text{vec } \mathbf{L}_{W(\mathbf{r}_t)}) + (\eta \text{vec } \mathbf{S} + \text{vec } \mathbf{V}) \right) \end{aligned} \quad (43)$$

□

Finally, the \mathbf{S}, \mathbf{V} updates are found in closed form.

3.3.3. \mathbf{S} -update

The update for \mathbf{S} can be expressed as:

$$\min_{\mathbf{S}} \quad \lambda \|\mathbf{M}\|_{2,1} + \frac{\eta}{2} \left\| \mathbf{S} - \mathbf{R} + \frac{1}{\eta} \mathbf{V} \right\|_F^2 \quad (44)$$

whose solution is a proximal of the $\ell_{2,1}$ -norm:

$$\mathbf{S} = T_{\lambda/\eta} \left(\mathbf{R} - \frac{1}{\eta} \mathbf{V} \right) \quad (45)$$

where T is the row thresholding operator.

3.3.4. \mathbf{V} -update

The \mathbf{V} -update is a generic ADMM step of dual ascent:

$$\mathbf{V} = \mathbf{V} + \eta(\mathbf{S} - \mathbf{R}). \quad (46)$$

Moreover, the dual balancing scheme [30] to adapt the dual hyper-parameters proved useful in practice. The method is summarized in Algorithm 3.

Algorithm 3 Algorithm for HKRPCA (full variable splitting)

- 1: Have: $\{\mathbf{y}_i\}_{i=1}^N, \{\Psi_i\}_{i=1}^N$
 - 2: Choose: $\lambda, \mu, \nu, \eta, c$ and \mathcal{P}
 - 3: $\mathbf{Y} \triangleq [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N]$
 - 4: $\Psi \triangleq [\Psi_1, \Psi_2, \dots, \Psi_N]$
 - 5: $\Psi_A \triangleq [\Psi_1^T \Psi_2^T \dots \Psi_N^T]^T$
 - 6: Initialize: $\mathbf{L}, \mathbf{R}, \mathbf{M}, \mathbf{S}, \mathbf{U}, \mathbf{V}, \mathbf{W}$
 - 7: **repeat**:
 - 8: $[\mathbf{L}]_{p_i} = \text{prox}_{(\mu/2\nu)H_c \circ \|\cdot\|_F} \left([\mathbf{M} + \frac{1}{\nu} \mathbf{U} - \mathbf{Y} + \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i} \right)$
 $+ [\mathbf{Y} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{p_i} \quad \forall p_i \in \mathcal{P}$
 - 9: **repeat**:
 - 10: $\mathbf{E} = \mathbf{Y} - \mathbf{L} - \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))$
 - 11: $[\mathbf{W}]_{j,k} = 1$ if $\|[\mathbf{E}]_{p_i}\|_F \leq c$ else $\sqrt{c / \|[\mathbf{E}]_{p_i}\|_F} \quad \forall (j, k) \in p_i$
 - 12: $\Psi_{AW} = \text{vec}(\mathbf{W}) \mathbf{1}^T \odot \Psi_A$
 - 13: $\Psi_{AWI} = \left(\frac{\mu}{2} \Psi_{AW}^H \Psi_{AW} + \eta \mathbf{I} \right)^{-1}$
 - 14: $\mathbf{r} = \Psi_{AWI} \left(\frac{\mu}{2} \Psi_{AW}^H (\text{vec } \mathbf{Y}_W - \text{vec } \mathbf{L}_W) + \frac{1}{\eta} \text{vec } \mathbf{S} + \text{vec } \mathbf{V} \right)$
 - 15: **until** stopping criterion is met
 - 16: $\mathbf{M} = D_{1/\nu}(\mathbf{L} - \frac{1}{\nu} \mathbf{U})$
 - 17: $\mathbf{S} = T_{\lambda/\eta}(\mathbf{R} - \frac{1}{\eta} \mathbf{V})$
 - 18: $\mathbf{U} = \mathbf{U} + \nu(\mathbf{M} - \mathbf{L})$
 - 19: $\mathbf{V} = \mathbf{V} + \eta(\mathbf{S} - \mathbf{R})$
 - 20: **until** stopping criterion is met
-

3.4. Convergence analysis

3.4.1. Semi-splitting algorithm

We consider the semi-splitting algorithm for HKRPCA, which we can write in the following equivalent formulation to (12):

$$\begin{aligned}
 \min_{\mathbf{L}, \mathbf{R}, \mathbf{M}} \quad & \|\mathbf{M}\|_* + \lambda \|\mathbf{R}\|_{2,1} + \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} H_c(\|\mathbf{S}_{p_i}(-\text{vec } \mathbf{Y} + \text{vec } \mathbf{L} + \Psi_A \text{vec } \mathbf{R})\|_F) \\
 \text{s.t.} \quad & \text{vec } \mathbf{M} - [\mathbf{I}_{MN}, \mathbf{0}_{MN \times N_x N_z R}] \begin{bmatrix} \text{vec } \mathbf{L} \\ \text{vec } \mathbf{R} \end{bmatrix} = \mathbf{0}_{MN}
 \end{aligned} \tag{47}$$

where \mathbf{S}_{p_i} denotes the selection matrix associated to the i^{th} block, which has a unique or no unit entry in each column/row and zeros elsewhere. $\mathbf{0}_{M \times N}$

denotes a matrix of zeros of M rows by M columns. Such a zeroes matrix acts on \mathbf{r} , as it is not split.

Then, the above problem may be cast in a 2-block ADMM with one composite variable $[\text{vec}(\mathbf{L})^T, \text{vec}(\mathbf{R})^T]^T$ with coefficient matrix $[\mathbf{I}_{MN}, \mathbf{0}_{MN} \Psi_A] = [\mathbf{I}_{MN}, \mathbf{0}_{MN \times N_x N_z R}]$ and associated composite convex objective function being the two latter terms of the objective function fused together.

In practice, solving directly over the composite variable is difficult so we solve for its sub-variables separately in a pass of Block Coordinate Descent (BCD), which is inexact and not part of the standard ADMM framework. Some works denoted Generalized ADMM (GADMM) [42] have been developed for approximate minimization but involve the introduction of a relaxation factor which changes the problem to solve.

We might think to cast the problem in a 3-block ADMM, which has been a topic of research the past few years [43, 44]: not necessarily convergent, a simple sufficient condition for its convergence is that any two coefficient matrices in the constraints must be orthogonal to each other. But, in our case, the objective function is not separable in the different components of the composite variable, so that we cannot apply the 3-block ADMM.

Thus, to the best of our knowledge, the analysis of the convergence of such a BCD split in a 2-block ADMM remains an open question while our experiments in the following Section 4 show its good practical recovery of the seeked result. The alternative use of GADMM may be investigated but will necessitate to solve new subproblems and to verify some additional sub-optimality conditions.

3.4.2. Full-splitting algorithm

In the case of a full split of variables i.e. splitting both \mathbf{L} and \mathbf{r} , we can rewrite the problem in the equivalent formulation:

$$\begin{aligned} \min_{\mathbf{L}, \mathbf{R}, \mathbf{M}, \mathbf{S}} \quad & \|\mathbf{M}\|_* + \lambda \|\mathbf{S}\|_{2,1} + \frac{\mu}{2} \sum_{p_i \in \mathcal{P}} H_c(\|\mathbf{S}_{p_i}(-\text{vec } \mathbf{Y} + \text{vec } \mathbf{L} + \Psi_A \text{vec } \mathbf{R})\|_F) \\ \text{s.t.} \quad & \begin{bmatrix} \text{vec } \mathbf{M} \\ \text{vec } \mathbf{S} \end{bmatrix} - \begin{bmatrix} \text{vec } \mathbf{L} \\ \text{vec } \mathbf{R} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{MN} \\ \mathbf{0}_{N_x N_z R} \end{bmatrix} \end{aligned} \tag{48}$$

we see that it lies in the realm of 2-block ADMM with two composite variables $[\text{vec}(\mathbf{L})^T, \text{vec}(\mathbf{R})^T]^T$ and $[\text{vec}(\mathbf{M})^T, \text{vec}(\mathbf{S})^T]^T$, with the latter term having associated objective function the sum of nuclear and $\ell_{2,1}$ norms.

Again, we only do inexact minimization over $[\text{vec}(\mathbf{L})^T, \text{vec}(\mathbf{R})^T]^T$ as well as for $[\text{vec}(\mathbf{M})^T, \text{vec}(\mathbf{S})^T]^T$ via Block Coordinate Descent (BCD). The question of its convergence is thus also open while experiments show good results.

3.5. Computational complexity

3.5.1. Semi-splitting algorithm

Assuming the same ordering of dimensions as for KRPCA, i.e. that $D > M > N$ where $D = N_x N_z R$ and that $MN > D$. The proximal of the Huber function composed with the Frobenius norm (plus a translation) is not the most costly operation as it scales linearly with the input matrix dimensions (so it is $\mathcal{O}(MN)$). The evaluation of $\Psi(\mathbf{I}_N \otimes \mathbf{r})$ is $\mathcal{O}(MND)$ as well as for the gradient evaluation in the PGD. Setting the number of PGD iterations to K , we have a computational complexity of $\mathcal{O}(KMND)$ for the algorithm.

3.5.2. Full-splitting algorithm

Via full splitting, so with a MM step for \mathbf{r} , we have the task of inverting a matrix at each MM iteration (or solving the associated linear system of equations) of size D which will be $\mathcal{O}(D^3)$ via Gaussian elimination. However, the major cost is the computation of the matrix product $\Psi_{AW}^H \Psi_{AW}$ inside the inverse, which will be $\mathcal{O}(NMD^2)$ and cannot be cached. This time again, consider K iterations of MM. Then, the cost of the \mathbf{r} -update via MM is $\mathcal{O}(KMND^2)$, which will be the overall computational complexity of the full splitting algorithm. Table 1 recapitulates the complexities of all algorithms proposed in this paper. We see the higher iteration cost of the full decoupling method compared to the semi-decoupling one.

Method	KRPCA	HRKRPCA SD	HKRPCA FD
Complexity	$\mathcal{O}(MND)$	$\mathcal{O}(KMND)$	$\mathcal{O}(KMND^2)$

Table 1: Computational complexity of the introduced methods

Figure 3 presents a study of the convergence speed of the different methods. In the point-block method, $\forall p_i \in \mathcal{P}$, p_i is the support of the i^{th} entry of \mathbf{Y} in some chosen order. We denote this setup for the semi-decoupling algorithm as HKRPCA SD-pt and HKRPCA FD-pt for the full-decoupling algorithm. In the column-block method, $\forall p_i \in \mathcal{P}$, p_i is the support of the i^{th} column \mathbf{y}_i . We denote this setup for the semi-decoupling algorithm as

HKRPCA SD-col and HKRPCA FD-col for the full-decoupling algorithm. It should be kept in mind that the different methods have different objective functions. Nevertheless, we see that their convergence in terms of iterations, except SRCS, behave similarly. Over time, we see that the point-wise HKRPCA methods (HKRPCA SD-pt and HKRPCA FD-pt) perform similarly albeit a bit slower than KRPCA, whereas their column-wise counterparts are noticeably slower (HKRPCA SD-col and HKRPCA FD-col). This is explained by the implementation: the point-wise application of the Huber function can be vectorized over the matrix, whereas the column-wise case necessitates the slicing of the matrix along the columns before applying the Huber function, which is computationally more demanding.

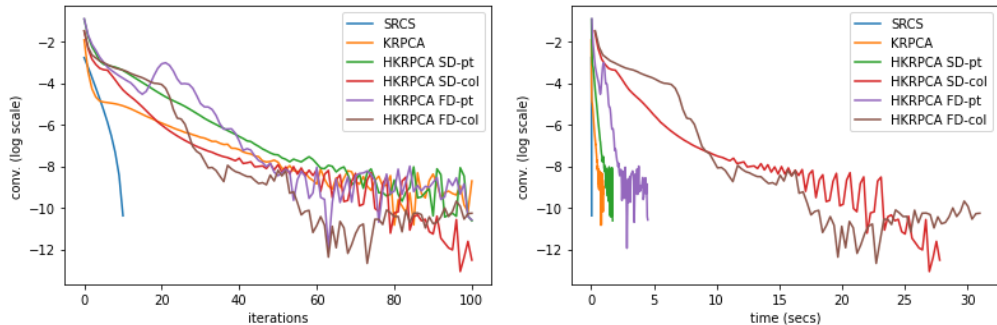


Figure 3: Convergence (log scale) vs iterations (left) and time (right)

4. Experiments

4.1. Simulation setup

4.1.1. FDTD data

We test our methods on electromagnetic simulations via Finite-Difference Time-Domain (FDTD) with GprMax [45]. The scene, as described in Figure 1, is 4.9×5.4 m in crossrange (x -axis) vs downrange (z -axis) with a discretization step of 3mm. The front wall, parallel to the SAR movement, is at a standoff distance to the radar of 1.2 m. It is homogeneous and non-conductive, of thickness 20cm and relative permittivity $\epsilon = 4.5$. One target is behind the wall, a perfect electric conductor (PEC) cylinder of radius 3mm situated at coordinates (2.6, 4). The radar moves 2cm along the x -axis between each acquisition, starting from $x = 1.824$ m, with 67 different positions overall. The signal sent is a ricker wavelet centered at 2 GHz.

4.1.2. Noise generation

To simulate different data acquisitions, we add random heterogeneous noise drawn from student-t noise. We will consider both pointwise and column wise noise. The column wise noise heterogeneity may arise as a result of the wall structure, e.g. drywall. The pointwise case may arise by adding the possibility of a frequency-dependant relative permittivity of the wall. Additionally, we consider the possibility of outliers coming from a different random process, which can be interpreted as mishandling in the acquisition process, etc.

We firstly consider two pointwise cases.

- pointwise noise only: $[\mathbf{Y}]_{i,j} = [\mathbf{L} + \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{i,j} + [\mathbf{T}]_{i,j}$
- pointwise noise + outliers: $[\mathbf{Y}]_{i,j} = [\mathbf{L} + \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{i,j} + [\mathbf{T}]_{i,j} + [\mathbf{O}]_{i,j}$

with $\mathbf{T}_{i,j}$ being i.i.d. centered univariate complex t-random variables with $f > 2$ degrees of freedom (d.f.) i.e. $\mathbf{T}_{i,j} \sim \mathcal{C}t_f(0, \sigma)$ where the standard deviation σ is adjusted to get the desired SNR level. \mathbf{O} is a matrix of outliers, whose number is set by the user and whose support Ω is randomly selected at uniform among all entries. The outliers are then drawn from a standard gaussian i.e. $\mathbf{O}_\Omega \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$. Entries of \mathbf{O} not in Ω are then set to zero.

Secondly we consider two column wise cases.

- column wise noise only: $\mathbf{y}_i = [\mathbf{L} + \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{:,i} + [\mathbf{T}]_{:,i}$
- column wise noise + outliers: $\mathbf{y}_i = [\mathbf{L} + \Psi(\mathbf{I}_N \otimes \text{vec}(\mathbf{R}))]_{:,i} + [\mathbf{T}]_{:,i} + [\mathbf{O}]_{:,i}$

where columns of \mathbf{T} are i.i.d. random variables drawn from a m -variate t-distribution: $\mathbf{T}_{:,i} \sim \mathcal{C}t_{m,f}(\mathbf{0}, \sigma\mathbf{I})$ with $f > 2$. The outlying columns are selected uniformly at random among all columns, with their support denoted Ω . The entries of \mathbf{O} on those columns then follow a standard gaussian distribution i.e. $\mathbf{O}_\Omega \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$ while entries not supported on Ω are set to zero.

4.1.3. Hyperparameter tuning

The hyperparameters have been tuned by hand in the following study. For fair comparisons, all algorithms are used with hyperparameters (when applicable): $\lambda = 1, \mu = 10, \nu = 1, c = 0.1, \eta = 1e10$, which have given good results for all methods. They are run the same number of iterations, as all algorithms iterations cycle through every variable, and a comparison in terms

of convergence is not possible, the methods converging based on different functionals. In order to avoid this tedious process, one may alternatively tune the hyperparameters using bayesian optimisation (see e.g. [46] and references therein). It uses a Gaussian Process (GP) prior over the f1-score of the detection map of the algorithm to tune. It is then possible to get an analytical formula for the posterior GP and to find sample hyperparameters to evaluate next based on some metric such as Expected Improvement. This can be readily implemented with the package BayesianOptimization [47].

The influence of the hyperparameters (λ, μ) on the performance of HKRPCA has been studied in Figure 4. There, each point’s Area Under the Curve (AUC) is averaged over 30 draws. We see that there is a fairly large range of values $\lambda \in [0, 20], \mu \in [1, 100]$ where the AUC is high. Additionally, we observed empirically that the Bayesian hyperparameter tuning method does propose values in this area (e.g. $\lambda = 14, \mu = 99$ here).

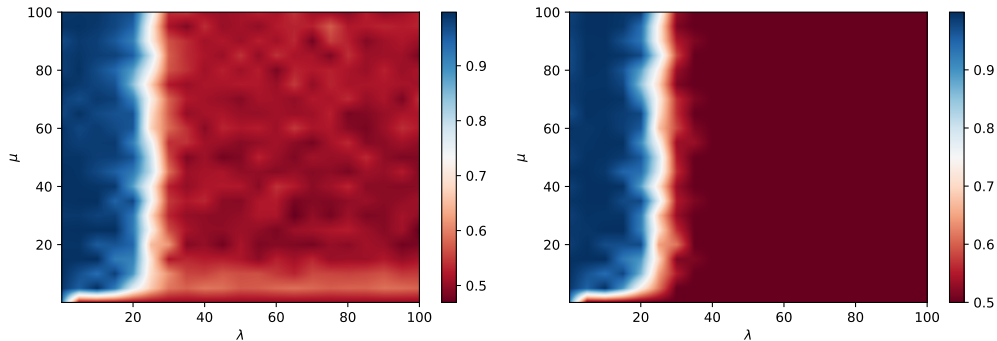


Figure 4: AUC over a grid of hyperparameters for HKRPCA FD-pt (left) and HKRPCA SD-pt (right) with pointwise noise

4.2. Performance evaluation

Sample results are shown for the different methods in Figure 5 with pointwise noise only. The target location is indicated with a red circle.

We evaluate quantitatively the performance of the methods based on their Receiver Operator Characteristic (ROC) averaged over 100 draws at each point of the curve.

4.2.1. Pointwise noise only

We begin with a setup consisting in only pointwise heterogeneous noise, that is following a centered multivariate student-t distribution. We chose

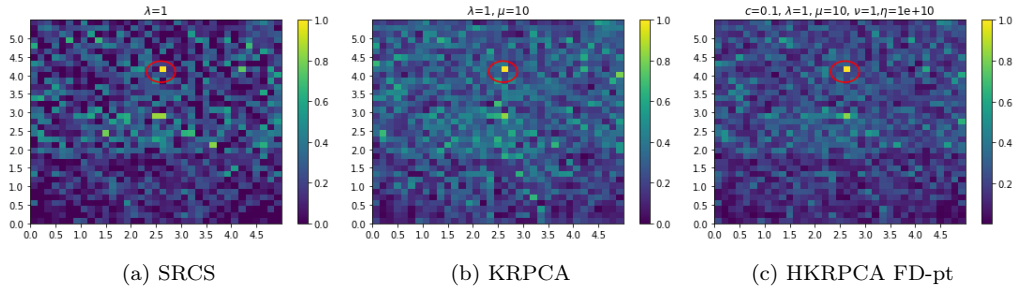


Figure 5: Sample detection maps (one target with location circled in red)

the setup of degrees of freedom: $\text{d.f.} = 2.01$ and Signal to Noise Ratio: $\text{SNR} = 10\text{dB}$ to visualize at best the difference in performance of the different methods. On Figure 6a, we plotted the resulting ROC. We observe that all methods with the Huber cost perform in a similar fashion. KRPCA performs worse and finally SRCS is the worst performing method.

4.2.2. Pointwise noise and point outliers

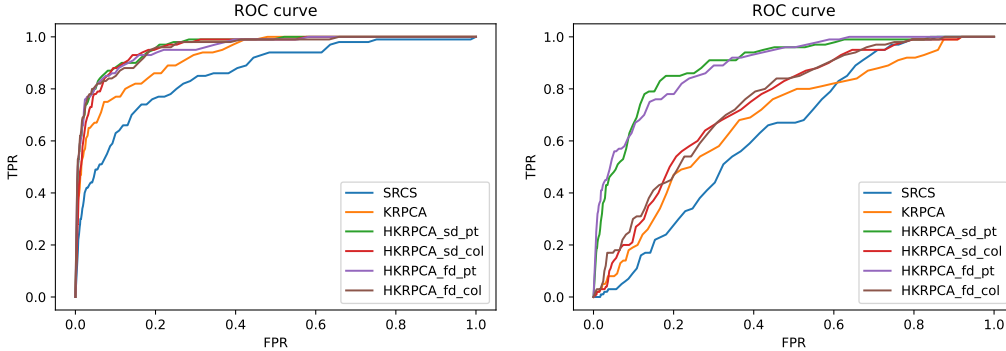
Next, we are interested in a setup with pointwise heterogeneous noise plus 100 point outliers, i.e. with perturbations coming from a different random process. Here the outlying entries have pointwise noise generated from a univariate standard gaussian distribution. On Figure 6b, we have the resulting ROC. We see that both HKRPCA SD-pt and HKRPCA FD-pt perform similarly and better than HKRPCA SD-col and HKRPCA FD-col. KRPCA and SRCS are the least well performing again.

4.2.3. Column wise noise only

To evaluate the effect of the block-wise methods, we thus generate block-wise noise to see its effects and the resulting discrepancy in performance of the different methods. On Figure 7a we have the resulting ROC with column-wise heterogeneous noise. In our setup, this means that the noise is considered radar position per radar position, and may change in power over radar acquisitions. We see here, with a bit more degraded setup than previous ones, that HKRPCA FD-col performs the best. Other methods except SRCS are a bit below on the graph, and SRCS is last.

4.2.4. Column wise noise and column outliers

For one last setup, we add outliers to the column-wise setup. To the column-wise heterogeneous noise, we add 25 column outliers i.e. with column-



(a) ROC with only pointwise heterogeneous noise (b) ROC with both pointwise noise and point outliers (student pointwise noise with d.f. = 2.01 and SNR= 10 dB) (student pointwise noise with d.f. = 2.1, SNR= 12 dB and 100 point outliers)

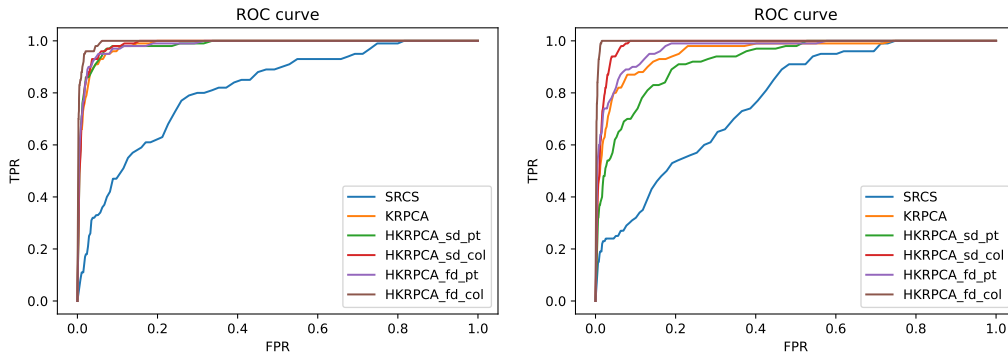
Figure 6: ROC with pointwise corruptions

wise noise generated from a standard multivariate gaussian. On Figure 7b we have the corresponding ROC. We see a clearer separation of performance between all methods. HRKPCA FD-col performs better than HKRPCA SD-col which in turns performs better than HRKPCA FD-pt. The method HRKPCA SD-pt comes after and KRPCA and SRCS are last.

On the whole, we have seen that the robust cost methods do perform better in heterogeneous noise scenarios, and that the correct block structure does impact the performance of those robust methods, especially and more clearly with outliers. Finally, the full decoupling method with a MM step performs better than the semi-decoupling method for blockwise setups.

5. Conclusion

In this paper, we stated a new method of one-step localisation of targets in the context of TWRI. It is designed to be robust to heterogeneous noise and outliers. The proposed resolution relies on the ADMM framework with two distinct algorithms tailored. On the one hand, a single split of the variable comprising the wall returns results in a closed form proximal evaluation. On the other hand, an additional split of the variable comprising the target returns lends itself to tailored MM step. We show on FDTD simulated data, in more complex scenarios where the noise is heterogeneous or outliers are present, that our method achieves better performances. This suggest further studies on real experimental data where the wall is not an idealized dielectric



(a) ROC with only column-wise heterogeneous noise (b) ROC with both column noise and column outliers (student columnwise noise with d.f. = 2.01 and liers (student columnwise noise with d.f. = 2.1, SNR= 6 dB) SNR= 12 dB and 25 column outliers)

Figure 7: ROC with column-wise corruptions

slab. Additionally, the methods proposed are in fact quite generic, and may be used in similar contexts such as Ground Penetrating Radar (GPR).

References

- [1] M. Amin, Through-the-Wall Radar Imaging, CRC Press, 2017.
- [2] Z. Li, T. Jin, Y. Dai, Y. Song, Through-wall multi-subject localization and vital signs monitoring using uwb mimo imaging radar, *Remote Sensing* 13 (2021).
- [3] D. Yang, Z. Zhu, J. Zhang, B. Liang, The overview of human localization and vital sign signal measurement using handheld ir-uwb through-wall radar, *Sensors* 21 (2021).
- [4] C. Debes, J. Hahn, A. M. Zoubir, M. G. Amin, Target discrimination and classification in through-the-wall radar imaging, *IEEE Transactions on Signal Processing* 59 (2011) 4664–4676.
- [5] C. Clemente, A. Balleri, K. Woodbridge, J. J. Soraghan, Developments in target micro-doppler signatures analysis: radar imaging, ultrasound and through-the-wall radar, *EURASIP Journal on Advances in Signal Processing* 2013 (2013) 1–18.
- [6] G. Gennarelli, G. Vivone, P. Braca, F. Soldovieri, M. G. Amin, Multiple extended target tracking for through-wall radars, *IEEE Transactions on Geoscience and Remote Sensing* 53 (2015) 6482–6494.
- [7] H. Li, G. Cui, L. Kong, G. Chen, M. Wang, S. Guo, Robust human targets tracking for mimo through-wall radar via multi-algorithm fusion, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 12 (2019) 1154–1164.

- [8] P. Protiva, J. Mrkvice, J. Machac, Estimation of wall parameters from time-delay-only through-wall radar measurements, *IEEE Transactions on Antennas and Propagation* 59 (2011) 4268–4278.
- [9] T. Jin, B. Chen, Z. Zhou, Image-domain estimation of wall parameters for autofocusing of through-the-wall sar imagery, *IEEE Transactions on Geoscience and Remote Sensing* 51 (2013) 1836–1843.
- [10] G. Wang, M. Amin, Imaging through unknown walls using different standoff distances, *IEEE Transactions on Signal Processing* 54 (2006) 4015–4025.
- [11] F. Ahmad, M. G. Amin, G. Mandapati, Autofocusing of through-the-wall radar imagery under unknown wall characteristics, *IEEE Transactions on Image Processing* 16 (2007) 1785–1795.
- [12] F. Ahmad, Multi-location wideband through-the-wall beamforming, in: 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, 2008, pp. 5193–5196.
- [13] M. Dehmollaian, K. Sarabandi, Refocusing through building walls using synthetic aperture radar, *IEEE Transactions on Geoscience and Remote Sensing* 46 (2008) 1589–1599.
- [14] M. Soumekh, a. O. M. C. Safari, *Synthetic Aperture Radar Signal Processing with MATLAB Algorithms*, Wiley-Interscience, 1999.
- [15] M. G. Amin, F. Ahmad, Compressive sensing for through-the-wall radar imaging, *Journal of Electronic Imaging* 22 (2013) 1 – 22.
- [16] P. K. Verma, A. N. Gaikwad, D. Singh, M. Nigam, Analysis of clutter reduction techniques for through wall imaging in uwb range, *Progress In Electromagnetics Research B* 17 (2009) 29–48.
- [17] F. H. C. Tivive, M. G. Amin, A. Bouzerdoum, Wall clutter mitigation based on eigenanalysis in through-the-wall radar imaging, in: 2011 17th International Conference on Digital Signal Processing (DSP), 2011, pp. 1–8.
- [18] F. H. C. Tivive, A. Bouzerdoum, M. G. Amin, A subspace projection approach for wall clutter mitigation in through-the-wall radar imaging, *IEEE Transactions on Geoscience and Remote Sensing* 53 (2015) 2108–2122.
- [19] Q. Huang, L. Qu, B. Wu, G. Fang, Uwb through-wall imaging based on compressive sensing, *IEEE Transactions on Geoscience and Remote Sensing* 48 (2010) 1408–1415.
- [20] M. Leigsnering, F. Ahmad, M. Amin, A. Zoubir, Multipath exploitation in through-the-wall radar imaging using sparse reconstruction, *IEEE Transactions on Aerospace and Electronic Systems* 50 (2014) 920–939.

- [21] E. J. Candès, X. Li, Y. Ma, J. Wright, Robust principal component analysis?, *Journal of the ACM (JACM)* 58 (2011) 1–37.
- [22] V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, A. S. Willsky, Rank-sparsity incoherence for matrix decomposition, *SIAM Journal on Optimization* 21 (2011) 572–596.
- [23] M. Mardani, G. Mateos, G. B. Giannakis, Recovery of low-rank plus compressed sparse matrices with application to unveiling traffic anomalies, *IEEE Transactions on Information Theory* 59 (2013) 5186–5205.
- [24] V. H. Tang, A. Bouzerdoum, S. L. Phung, F. H. C. Tivive, Radar imaging of stationary indoor targets using joint low-rank and sparsity constraints, in: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 1412–1416.
- [25] V. H. Tang, A. Bouzerdoum, S. L. Phung, Compressive radar imaging of stationary indoor targets with low-rank plus jointly sparse and total variation regularizations, *IEEE Transactions on Image Processing* 29 (2020) 4598–4613.
- [26] A. Breloy, M. N. El Korso, A. Panahi, H. Krim, Robust subspace clustering for radar detection, in: *2018 26th European Signal Processing Conference (EUSIPCO)*, 2018, pp. 1602–1606.
- [27] B. Mériaux, A. Breloy, C. Ren, M. N. El Korso, P. Forster, Modified sparse subspace clustering for radar detection in non-stationary clutter, in: *2019 IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2019, pp. 669–673.
- [28] H. Brehier, A. Breloy, C. Ren, I. Hinojosa, G. Ginolhac, Robust pca for through-the-wall radar imaging, in: *2022 30th European Signal Processing Conference (EUSIPCO)*, IEEE, 2022, pp. 2246–2250.
- [29] P. J. Huber, Robust Estimation of a Location Parameter, *The Annals of Mathematical Statistics* 35 (1964) 73 – 101.
- [30] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, *Found. Trends Mach. Learn.* 3 (2011) 1–122.
- [31] A. Aravkin, S. Becker, V. Cevher, P. Olsen, A variational approach to stable principal component pursuit, *Uncertainty in Artificial Intelligence - Proceedings of the 30th Conference, UAI 2014* (2014).
- [32] F. Ahmad, M. G. Amin, Noncoherent approach to through-the-wall radar localization, *IEEE Transactions on Aerospace and Electronic Systems* 42 (2006) 1405–1419.
- [33] R. Tibshirani, Regression shrinkage and selection via the Lasso, *Journal of the Royal Statistical Society. Series B (Methodological)* 58 (1996) 267–288.

- [34] M. Kowalski, Sparse regression using mixed norms, *Applied and Computational Harmonic Analysis* 27 (2009) 303–324.
- [35] M. Fazel, Matrix rank minimization with applications, Ph.D. thesis, PhD thesis, Stanford University, 2002.
- [36] N. Parikh, S. Boyd, et al., Proximal algorithms, *Foundations and Trends® in Optimization* 1 (2014) 127–239.
- [37] E. Ollila, D. E. Tyler, V. Koivunen, H. V. Poor, Complex elliptically symmetric distributions: Survey, new results and applications, *IEEE Transactions on Signal Processing* 60 (2012) 5597–5625.
- [38] R. Maronna, R. Martin, V. Yohai, M. Salibián-Barrera, *Robust Statistics: Theory and Methods (with R)*, Wiley Series in Probability and Statistics, Wiley, 2019.
- [39] A. Beck, *First-Order Methods in Optimization*, SIAM-Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2017.
- [40] Y. Sun, P. Babu, D. Palomar, Majorization-minimization algorithms in signal processing, communications, and machine learning, *IEEE Trans. on Signal Process.* PP (2016) 794–816.
- [41] J. de Leeuw, K. Lange, Sharp quadratic majorization in one dimension, *Computational Statistics and Data Analysis* 53 (2009) 2471–2484.
- [42] E. Fang, B.-S. He, H. Liu, X. Yuan, Generalized alternating direction method of multipliers: New theoretical insights and applications, *Mathematical Programming Computation* 7 (2015).
- [43] D.-R. Han, A survey on some recent developments of alternating direction method of multipliers, *Journal of the Operations Research Society of China* (2022) 1–52.
- [44] C. Chen, B. He, Y. Ye, X. Yuan, The direct extension of admm for multi-block convex minimization problems is not necessarily convergent, *Math. Program.* 155 (2016) 57–79.
- [45] C. Warren, A. Giannopoulos, I. Giannakis, gprMax: Open source software to simulate electromagnetic wave propagation for ground penetrating radar, *Computer Physics Communications* 209 (2016) 163–170.
- [46] J. Snoek, H. Larochelle, R. P. Adams, Practical bayesian optimization of machine learning algorithms, in: F. Pereira, C. Burges, L. Bottou, K. Weinberger (Eds.), *Advances in Neural Information Processing Systems*, volume 25, Curran Associates, Inc., 2012.
- [47] F. Nogueira, *Bayesian Optimization: Open source constrained global optimization tool for Python*, 2014–.