2018 5th International Workshop on Compressed Sensing applied to Radar, Multimodal Sensing, and Imaging (CoSeRa)

Learning and Exploiting Physics of Degradations

Paul Escande
Department of Applied Mathematics and Statistics
Johns Hopkins University
Baltimore, Maryland 21218
Email: paul.escande@gmail.com

Mauro Maggioni
Department of Mathematics
Department of Applied Mathematics and Statistics
Mathematical INstitute of Data Science (MINDS)
Institute of Data Intensive Engineering and Science (IDIES)
Johns Hopkins University
Baltimore, Maryland 21218
Email: mauro.maggioni@jhu.edu

***Abstract*—Even though physics of degradations of an acquisition system might be complex, it often relies on a small number of parameters. We consider the blind deconvolution problem with a kernel $k(\theta)$ parametrized by $\theta \in \Theta \subset \mathbb{R}^m$. Assuming that m is small, this setting models the dependency of physics of degradations on a *small number of parameters*. Under some hypotheses on Θ and the map $\theta \mapsto k(\theta)$, the set $\mathcal{M} = \{k(\theta) \,|\, \theta \in \Theta\}$ describes a compact Riemannian *manifold*. This works deals with *learning* the manifold $\mathcal{M}$ and its *exploitation* in restoration methods.**

I. Introduction

We consider the following blind deconvolution problem:

$$u_0 = k(\theta) \star u + \eta, \tag{1}$$

where $u_0 \in R^N$ is the observed image, $u \in \mathbb{R}^N$ is the underlying sharp image, $\eta \sim \mathcal{N}(0, \sigma^2 \mathrm{Id}_N)$ is an additive noise and $k(\theta) \in \mathbb{R}^K$ is a convolution kernel parametrized by $\theta \in \Theta \subset \mathbb{R}^m$. We will denote $k : \Theta \to \mathbb{R}^K$ the parametrization of the kernels.

We assume Θ is embedded with the Euclidean distance and a measure μ that is absolutely continuous w.r.t. the volume measure. The set $\mathcal{M} = k(\Theta)$ is embedded with the L^2 metric and the push-forward measure $\nu = k\sharp\mu$. We assume that $(\mathcal{M}, \rho, \nu)$ is a compact $C^{1+\alpha}$ Riemannian manifold of dimension m isometrically embedded in $\mathbb{R}^K$. For example, this happens when Θ is an open hyperectangle of Θ and k is a $C^{1+\alpha}$ bi-Lipschitz map.

The classical blind deconvolution problem

$$\text{Find } (k, u) \text{ s.t. } u_0 = k \star u + \eta, \tag{2}$$

is extremely challenging to solve because of i) its ill-posedness, ii) its non-convexity due to the joint estimation (k, u) and iii) the number of degrees of freedom K can become large. Probably dating from 1975 [1], this problem has drawn a lot of attention in the image processing community in the last decades [2], [3], [4]. Most of these works rely on the minimization of a cost function of kind

$$\frac{1}{2}\|k \star u - u_0\|_2 + \lambda R_1(u) + \mu R_2(k), \tag{3}$$

via alternate minimizations. The optimization scheme is paired with an extensive number of empirical rules to (i) deal with the non-convexity of the problem and (ii) select a good local minima since the global minimizer of the cost function is oftentimes the trivial solution (δ, u_0) [5]. Even without reconstruction guarantees, these methods surprisingly provide excellent results.

Recently, some works have followed another path [6], [7], [8]. Based on the lifting trick, the bilinear problem becomes linear at the cost of an increase of its dimension. The problem boils down to recovering a rank one matrix. Even though it is in general NP-hard, it has been shown that under appropriate random measurement model, recovery guarantees can be established with high probability. In [6], a convex relaxation involving the minimization of the nuclear norm has been proposed. While providing a straightforward convergence of the algorithm to the global solution, this strategy becomes intractable for large dimensions. To copt with this issue, [7] proposed to solve the non-convex problem. By choosing a wise initial guest, the algorithm is shown to converge to the solution of the problem with high probability. From a practical point of view, those methods require a rather precise estimation of the support of the kernel to output good restoration results.

These existing works might not be tailored to the setting (1), where the kernels are supposed to leave on a manifold of small intrinsic dimension. They could be improved be improved taking advantage of the additional manifold structure. Furthermore, it is of great interest to design efficient methods allowing to provide recovery guarantees while using state of the art image processing algorithm.

Assuming that the domain Θ can be sampled and the corresponding kernel can be estimated, we present a method that i) learns the manifold and ii) use this knowledge in a restoration procedure. After exposing notation, the method is described and then numerically illustrated on simulated data.

II. Notation

Let E denote the vector space of images defined on $\Omega = \{1, \dots, n_1\} \times \dots \times \{1, \dots, n_d\}$. The total number of pixels is therefore $N = n_1 \times \dots \times n_d$. The pixels of the image are identified by a multi-index $i = (i_1, \dots, i_d) \in \Omega$. For an image $u \in E$ and scalar $q \in [1, +\infty)$, we let $\|u\|_q = \left(\sum_{i\in\Omega} |u[i]|^q\right)^{1/q}$ and $\|u\|_\infty = \max_{i\in\Omega} |u[i]|$ denote the standard ℓ^q and ℓ^∞ norms, respectively. The dot product of u and $v \in E$ is defined by $\langle u, v\rangle = \sum_{i\in\Omega} u[i]v[i]$. Let $V = E^d$ denote the space of discrete vector field on Ω. For any $f = (f_1, \dots, f_d) \in V$, $|f|$ denotes an element of E with

i-th entry defined by

$$|f|[i] = \sqrt{\sum_{l=1}^{d} f_l[i]^2}. \tag{4}$$

The discrete gradient of $u \in E$ is defined by:

$$\nabla u = (\partial_1 u, \dots, \partial_d u) \in V. \tag{5}$$

Partial derivatives are defined by:

$$(\partial_l u)[i] = \begin{cases} u[\dots, i_l + 1, \dots] - u[\dots, i_l, \dots] & \text{if } i_l < n_l \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

The adjoint operator ∂_l^* of ∂_l is the unique operator satisfying $\langle \partial_l u, f_l \rangle = \langle u, \partial_l^* f_l \rangle$ for all $u, f_l \in E$. The adjoint of the gradient operator by $\nabla^* : V \to E$ is defined by $q \mapsto \sum_{l=1}^{d} \partial_l^* q_l$.

Let f be a function from $\mathbb{R}^m \to \mathbb{R}$, its gradient will be denoted as ∇f. For vector-valued function $g : \mathbb{R}^m \to \mathbb{R}^n$, ∇g refers to its Jacobian. Furthermore, if a function $h : \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} \to \mathbb{R}^n$ depends on two variables (x, y), $\nabla_x g$ will denote its gradient (or Jacobian) w.r.t. x.

III. Description of the method

A. Overview

The main idea of this work is to take advantage of the manifold $(\mathcal{M}, \rho, \nu)$ in restoration methods. In practice this manifold is never known and only samples from ν can be drawn i.i.d.

The method is therefore decomposed in two steps:

1) Construct an estimation $\widehat{\mathcal{M}}$ of the manifold $\mathcal{M}$ from a set of $(\widehat{k_i})$. This step is handled assuming that one can design a device able to sample $(\theta_i)_{i=1}^p \in \Theta$ i.i.d following μ. We also assume that one can construct a procedure giving $\widehat{k_i}$ an estimation of $k(\theta_i)$.
2) Minimize a cost function defined on this manifold. In this work we chose to solve the bi-level optimization problem:

$$\begin{cases} \min\limits_{k \in \widehat{\mathcal{M}}} \mathcal{J}(F(k)) = J(k) \\ F(k) = \arg\min\limits_{u \in E} \dfrac{1}{2} \|k \star u - u_0\|_2^2 + \lambda R(u). \end{cases} \tag{7}$$

The choices of the functions $\mathcal{J} : \mathbb{R}^N \to \mathbb{R}^+$ and $R : \mathbb{R}^N \to \mathbb{R}^+$ are discussed in the following sections.

B. Manifold learning

Many manifold learning techniques have been proposed e.g. [9], [10], [11]. We will use the Geometric Multi-Resolution Analysis (GMRA) [12] since it provides a robust and computationally efficient procedure to construct low-dimensional geometric approximations of $\mathcal{M}$ at varying resolutions from a i.i.d. sampling of ν.

The construction of GMRA involves the following steps:

1) Construct a multiscale partition $\{C_{j,i}\}_{j \in \mathbb{Z}, i \in \Lambda_j}$ of $(\mathcal{M}, \rho, \nu)$.
2) Perform local PCA on each $C_{j,i}$. Let $c_{j,i}$ be the mean of $C_{j,i}$ and $V_{j,i}$ be the m-dim principal subspace. Define $\mathcal{P}_{j,i}(x) = c_{j,i} + \text{Proj}_{V_{j,i}}(x - c_{j,i})$.
3) Get an approximation of $\mathcal{M}$ at scale j with $\mathcal{M}_= \{\mathcal{P}_{j,i}(C_{j,i})\}_{i \in \Lambda_j}$. Furthermore let $\mathcal{P}_j : \mathbb{R}^K \to \mathbb{R}^K$ denote the projection on $\mathcal{M}_j$.

In practice, the measure ν is never known and only samples $(k_i)_{i=1}^p$ can be sample from it i.i.d. The GMRA can be run with $\widehat{\nu}_p = \frac{1}{p} \sum_{i=1}^p \delta_{k_i}$ outputting a $\widehat{\mathcal{M}}_j = \{\widehat{\mathcal{P}}_{j,i}(C_{j,i})\}_{i \in \Lambda_j}$ and the associated projection $\widehat{\mathcal{P}}_j$.

The following Theorem gives guarantees on the empirical approximation of $\mathcal{M}$ by $\widehat{\mathcal{M}}_j$.

Theorem 3.1 ([13, Theorem 4]): Let $\mathcal{M}$ be a closed manifold of class $C^{1+\alpha}$ isometrically embedded in $\mathbb{R}^K$, with $\alpha \in (0, 1]$ and ν be a doubling probability measure on $\mathcal{M}$. Let $a > 0$ be arbitrary and $b > 0$. Let j^* be chosen such that

$$2^{-j^*} \propto \begin{cases} \dfrac{\log p}{p}, & \text{for } m = 1 \\ \left(\dfrac{\log p}{p}\right)^{\frac{1}{2(1+\alpha)+m-2}}, & \text{for } m \geq 2. \end{cases} \tag{8}$$

then

$$\mathbb{E}\|X - \widehat{\mathcal{P}}_{j^*} X\|^2 \lesssim \begin{cases} \left(\dfrac{\log p}{p}\right)^2, & \text{for } m = 1 \\ \left(\dfrac{\log p}{p}\right)^{\frac{2(1+\alpha)}{2(1+\alpha)+m-2}}, & \text{for } m \geq 2, \end{cases} \tag{9}$$

Theorem 3.1 also provides concentration bounds on $\|X - \widehat{\mathcal{P}}_j X\|$ that we do not expose for sake of space, see [13, Theorem 4] for more details. This result shows that GMRA has an estimation rate that depends exponentially only on m, the intrinsic dimension of the manifold, and not K, the ambient one. Furthermore, the numerical costs of computing $\widehat{\mathcal{M}}_j$ behaves like $O(pK(\log p + m^2)) + O_{m,K}(p \log p)$ where $O_{m,K}$ has a constant exponential in m and linear in K. These two properties illustrate the efficiency of GMRA in the targeted setting $m \ll K$.

C. Optimization on GMRA

The first idea of this work was to design a smooth cost function J on the GMRA which will be minimized by a gradient descent.

We will assume that R is chosen such that the $\arg\min$ in (7) is uniquely defined. If the maps $\mathcal{J} : E \to \mathbb{R}^+$ and $F : \mathbb{R}^K \to E$ are C^1 then so is $J : \mathbb{R}^K \to \mathbb{R}^+$. Furthermore its gradient reads $\nabla J(k) = (\nabla F(k))^* \nabla \mathcal{J}(F(k))$ where $\nabla F(k)$ denotes the Jacobian of F.

A generic gradient descent algorithm on a generic manifold $\mathcal{M}$ is as follows [14], [15]:

1) Compute extrinsic gradient $\nabla J(k^{(l)})$.
2) Projection: $\nabla_{\mathcal{M}, k^{(l)}} J(k^{(l)}) = P_{\mathcal{M}, k^{(l)}}(\nabla J(k^{(l)}))$, where $P_{\mathcal{M}, k^{(l)}}$ is the projection on the tangent plane to $\mathcal{M}$ at $k^{(l)}$.
3) Compute step-size $\alpha^{(l)}$ along the descent direction.

4) Retraction: $k^{(l+1)} = R_{\mathcal{M},k^{(l)}}(-\alpha^{(n)}\nabla_{\mathcal{M},k^{(l)}}J(k^{(l)}))$.

The GMRA structure allows some simplification of the algorithm above. Since it builds a set of planes that approximate the manifold, the tangent plane to $\mathcal{M}_j$ at $k^{(l)}$ is $V_{j,i}$ associated to $k^{(l)}$. Furthermore, the retraction operator is in this case the projection on the GMRA $\mathcal{P}_{j,i}$ associated to $k^{(l)}$.

D. Choice of R

For a fixed kernel k, the problem of deblurring has been extensively studied in the literature. The assumption that sharp images have small total variation has lead to various deblurring algorithms [16], [3].

However the total-variation is non-smooth and requires the use of complex algorithms, i.e. with a more involved differentiation step. For sake of investigation, we will instead use the smoothed isotropic total variation TV_ϵ of an image $u \in E$ defined as:

$$TV_\epsilon(u) = \sum_{i\in\Omega} \sqrt{|\nabla u|[i]^2 + \epsilon^2}. \tag{10}$$

The parameter ϵ allows to smooth the singularity of the regular total-variation that is recovered when $\epsilon = 0$.

The nested optimization procedure to solve is therefore:

$$F(k) = \arg\min_{u\in E} \frac{1}{2}\|k\star u - u_0\|_2^2 + \lambda TV_\epsilon(u) = G(u,k). \tag{11}$$

In practice $F(k)$ is never reached and we are always given an procedure F that approximately solves (11). In the following, we make the abuse of notation that F is the optimization procedure e.g. F can be a gradient-descent scheme of L iterations. In this case $F(k)$ gives an approximation of u^* the minimizer of (11).

E. Sharpness index

The choice of $\mathcal{J}$ is also crucial. it has been shown in [5] that most of the usual regularizers used in [2], [3], [4] - e.g. total variation, sparsity in wavelet domain - suffer from the pitfall of the trivial solution i.e. the global minimizer of J is the projection of the identity on the manifold of kernels. To copt out this issue we chose to use a measure of phase coherence of the image.

The sharpness index has been introduced in [17] as a measure of the Global Phase Coherence (GPC) of an image. The key idea behind this notion is that the phase coefficients of an image encode its geometry. The global coherence measures how much this geometry varies when the phase information is perturbed. Let $\widehat{u_\psi}(\xi) = |\widehat{u}(\xi)|e^{i\psi(\xi)}$ denote the randomly perturbed image with a uniform odd random phase function ψ. The geometry of an image can be characterized by its total variation. This leads to the following definition of global phase coherence:

$$GPC(u) = -\log_{10}\mathbb{P}(TV(u_\psi) \le TV(u)). \tag{12}$$

The bigger the GPC is, the smaller is the probability of the TV to decrease by phase perturbations, meaning that the image is sharper.

The critical issue of the GPC is that there is no closed form formula to evaluate (12). This issue can be circumvented by considering a relaxation of the random image model $\widehat{u_\psi}$ by $\widehat{u\star W} = |\widehat{u}(\xi)||\widehat{W}(\xi)|e^{i(\arg\widehat{u}+\arg\widehat{W})}$ where W is a Gaussian white noise with standard deviation $N^{-1/2}$. Here, both the modulus and the phase of the image are perturbed. It can be shown that those two image models are close see [17]. We define the sharpness index as:

$$SI(u) = -\log_{10}\mathbb{P}(TV(u\star W) \le TV(u)). \tag{13}$$

The main advantage of using SI is that $u\star W$ is approximately Gaussian - see again [17] for asymptotic results. Therefore, SI can be evaluated as

$$SI(u) = -\log_{10}\Phi\left(\frac{\mu - TV(u)}{\sigma}\right), \tag{14}$$

where $\Phi(t) = \int_t^\infty e^{-s^2/2}ds$ is the Gaussian tail and $\mu = \mathbb{E}[TV(u\star W)]$ and $\sigma^2 = \mathrm{Var}(TV(u\star W))$. The mean μ and the variance σ^2 can be computed explicitly as functions of u using a few number of FFTs [17].

Two problems arise with the use of such function. First, the SI is neither concave or convex. This will require to input good initialization of the gradient descent on the manifold. This can be done exploiting the coarse to fine structure of the GMRA. Second, it is non-smooth. To overcome this difficulty, we will use the SI_ϵ defined with TV_ϵ instead of TV in (14).

F. Bi-level optimization

In this subsection, we are interested in computing the extrinsic gradient of J in (7) with $R = TV_\epsilon$ and $\mathcal{J} = SI$. This gradient involves the derivative of the solution given by the algorithm with respect to the kernel. This idea has been similarly investigated in [18], [19].

Problem (11) can be solved using a gradient descend:

- $u^{(l+1)} = u^{(l)} - \tau\nabla_u G(u^{(l)}, k)$.

The convergence of $G(u^{(l)},k)$ to $G(u^*,k)$ is guaranteed as long as $\tau < 2/L$ where L is the Lipschitz constant of $\nabla_u G$. Moreover, if G is a C^2 strongly convex function with M-Lipschitz Hessian, then the convergence of $u^{(l)}$ to u^* is ensured in a linear rate provided some choice on the step-size and that the initial guess $u^{(0)}$ is closed enough of the solution [20]. The gradient descent can be accelerated using some acceleration like Nesterov's ones [20].

For gradient descent schemes with L iterations, the mapping F is defined by $k \mapsto u^{(L)}$. We are interested in computing the Jacobian of $u^{(L)}$ with respect to k obtained through the iterative algorithm $u^{(l+1)} = \psi(u^{(l)},k) = u^{(l)} - \tau\nabla_u G(u^{(l)},k)$. By commodity, we let F_l denote the map $k \mapsto u^{(l)}$. The Jacobian $\nabla F(k) = \nabla F_L(k)$ can be obtained recursively through the chain rule

$$\begin{aligned}\nabla F_{l+1}(k) &= \nabla_u\psi(u^{(l)},k)\nabla F_l(k) + \nabla_k\psi(u^{(l)},k)\\ &= (\mathrm{Id} - \tau\nabla_u\nabla G(u))\nabla F_l(k) - \tau\nabla_k\nabla G(u)\end{aligned} \tag{15}$$

The Jacobian $\nabla F_l(k)$ is of size $N\times K$, $\nabla_u\psi(u^{(l)},k)$ is of size $N\times N$ and $\nabla_k\psi(u^{(l)},k)$ of size $N\times K$. This matrices can be huge for images commonly encountered in applications.

However these matrices can be efficiently applied to arbitrary vectors with convolutions. The computation of the gradient is therefore of the order same as one evaluation of F.

IV. Numerical experiments

In this section, we will study the behavior of the method on simulated data. In this setting, we use a skewed Gaussian model in 1d of the form:

$$h(t) \propto e^{-\frac{t^2}{2\alpha^2}} \int_{-\infty}^{\beta t} e^{-\frac{s^2}{2}} ds, \tag{16}$$

where $\alpha > 0$ and $\beta \in \mathbb{R}$ are respectively the variance and the shape parameters. Then, $p = 2,000$ sets of parameters are i.i.d. sampled from μ an uniform probability distribution on the product space $[1, 5] \times [-3/2, 3/2]$. We let $\{k_i\}_{i=1}^p$ denote the associated kernels.

The GMRA algorithm is run on these kernels and outputs a multi-scale approximation of the manifold. A new kernel k_0 is picked at random from μ and generate the degraded signal $u_0 = k \star u + \eta$ where η is a Gaussian white noise.

Finally we run the gradient descent algorithm on the fifth scale of the GMRA. With $L = 20$ iterations and starting point the center of the root cell, we obtain the results displayed in Figure 1.

V. Conclusion and future works

We proposed a numerical method allowing to learn the physic of degradations (1) i.e. estimating a manifold of small intrinsic dimension. We also designed a restoration method taking advantage of this extra knowledge.

Many methodological and theoretical questions have not been addressed yet. Other functions R and $\mathcal{J}$ can be investigated. Especially, it would be of great interest to chose them so that J can allow to derive recovery guarantees as [6], [7], [8]. Furthermore, the performances of the gradient descent on the manifold should be studied when run on the estimation $\widehat{\mathcal{M}}$ given on the GMRA.

The final goal of this method is to be used for spatially varying blur operators. It will drastically reduce the number of parameters to estimate from N^2 to m.

References

[1] T. G. Stockham, T. M. Cannon, and R. B. Ingebretsen, "Blind deconvolution through digital signal processing," *Proceedings of the IEEE*, vol. 63, no. 4, pp. 678–692, 1975.

[2] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[3] T. F. Chan and C.-K. Wong, "Total variation blind deconvolution," *IEEE transactions on Image Processing*, vol. 7, no. 3, pp. 370–375, 1998.

[4] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1964–1971.

[5] A. Benichoux, E. Vincent, and R. Gribonval, "A fundamental pitfall in blind deconvolution with sparse and shift-invariant priors," in *ICASSP-38th International Conference on Acoustics, Speech, and Signal Processing-2013*, 2013.

[6] A. Ahmed, B. Recht, and J. Romberg, "Blind deconvolution using convex programming," *IEEE Transactions on Information Theory*, vol. 60, no. 3, pp. 1711–1732, 2014.

[7] X. Li, S. Ling, T. Strohmer, and K. Wei, "Rapid, robust, and reliable blind deconvolution via nonconvex optimization," *arXiv preprint arXiv:1606.04933*, 2016.

[8] P. Jung, F. Krahmer, and D. Stöger, "Blind demixing and deconvolution at near-optimal rate," *arXiv preprint arXiv:1704.04178*, 2017.

[9] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[10] D. L. Donoho and C. Grimes, "Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data," *Proceedings of the National Academy of Sciences*, vol. 100, no. 10, pp. 5591–5596, 2003.

[11] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker, "Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 21, pp. 7426–7431, 2005.

[12] W. K. Allard, G. Chen, and M. Maggioni, "Multi-scale geometric methods for data sets ii: Geometric multi-resolution analysis," *Applied and Computational Harmonic Analysis*, vol. 32, no. 3, pp. 435–462, 2012.

[13] W. Liao and M. Maggioni, "Adaptive geometric multiscale approximations for intrinsically low-dimensional data," *arXiv preprint arXiv:1611.01179*, 2016.

[14] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[15] A. Kovnatsky, K. Glashoff, and M. M. Bronstein, "Madmm: a generic algorithm for non-smooth optimization on manifolds," in *European Conference on Computer Vision*. Springer, 2016, pp. 680–696.

[16] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.

[17] A. Leclaire and L. Moisan, "No-reference image quality assessment and blind deblurring with sharpness metrics exploiting fourier phase information," *Journal of Mathematical Imaging and Vision*, vol. 52, no. 1, pp. 145–172, 2015.

[18] C.-A. Deledalle, S. Vaiter, J. Fadili, and G. Peyré, "Stein unbiased gradient estimator of the risk (sugar) for multiple parameter selection," *SIAM Journal on Imaging Sciences*, vol. 7, no. 4, pp. 2448–2487, 2014.

[19] P. Ochs, R. Ranftl, T. Brox, and T. Pock, "Bilevel optimization with nonsmooth lower level problems," in *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer, 2015, pp. 654–665.

[20] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2013, vol. 87.

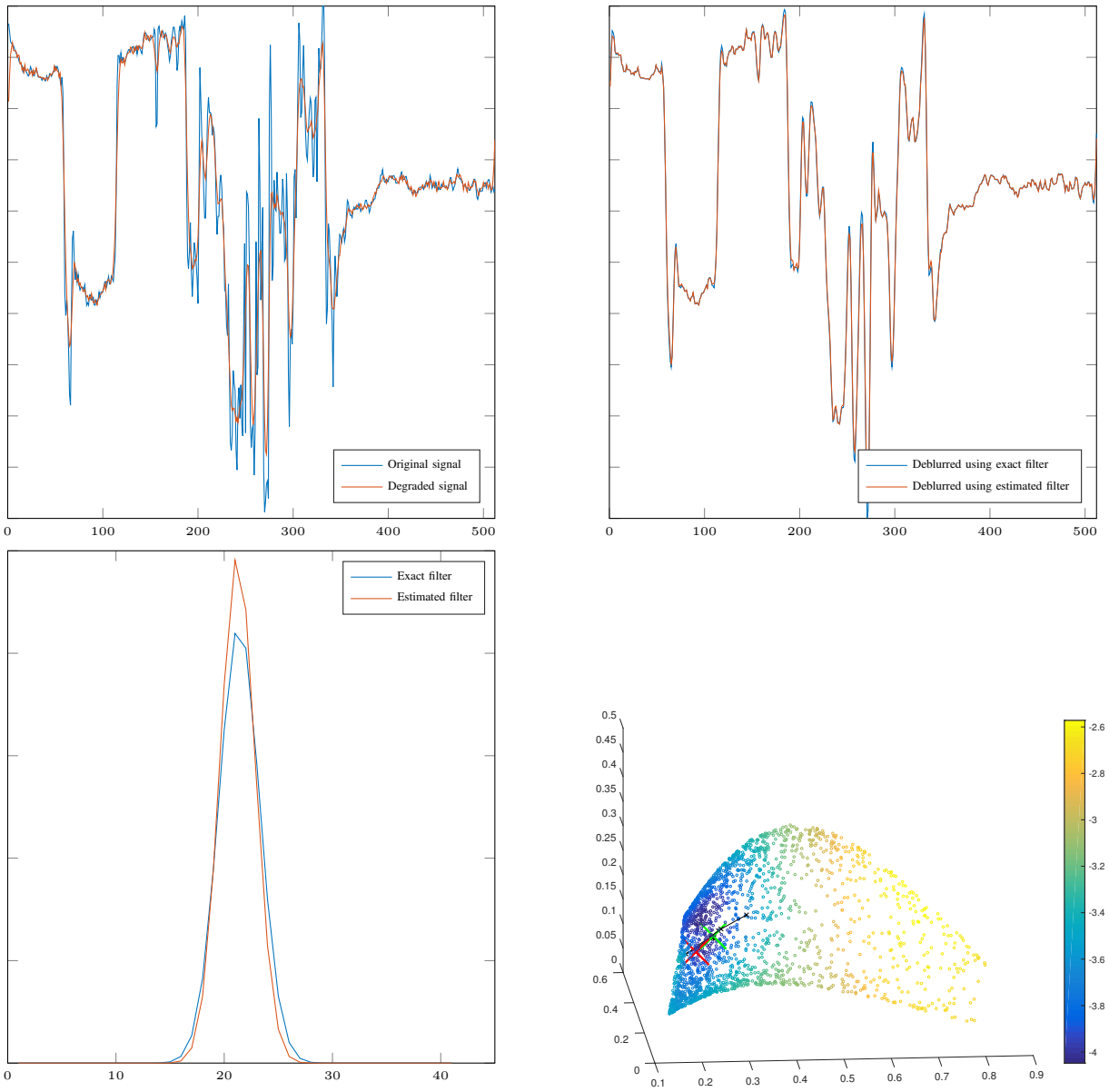

Fig. 1: Illustration of the output of the gradient descent algorithm. Top left: the original signal and its degraded version. Bottom left: the exact kernel k_0, and the estimated one. Top right: the restored signals using the exact filter and the estimated one. Bottom right: the manifold of the three first coordinates. The points are colored w.r.t. the value of the cost function. The red cross shows the location of h_0, and the green one of the estimated kernel. The black path shows the iterates of the gradient descent.