Generalized Inertial Proximal Deblurring

YANN SAVOYE, University of Leicester, United Kingdom
DAMRONGSAK YAMBANGWAI, University of Phayao, Thailand
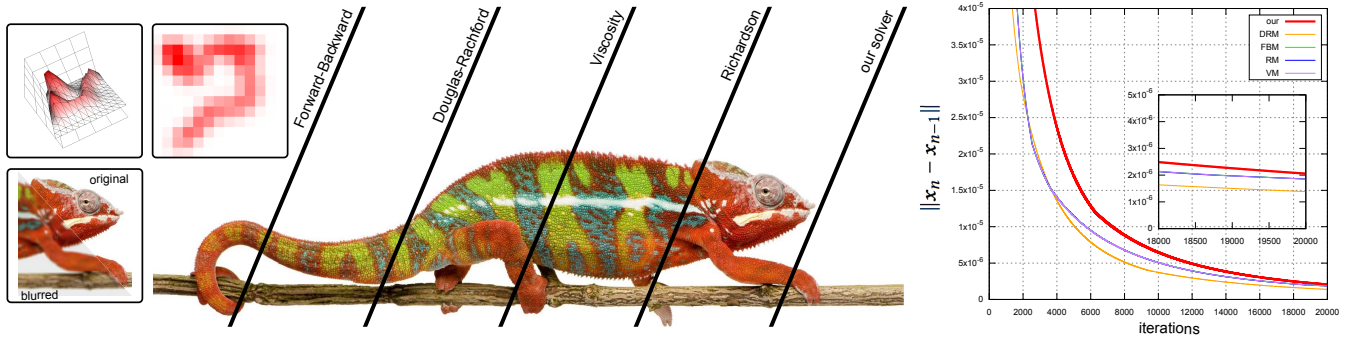WATCHARAPORN CHOLAMJIAK, University of Phayao, Thailand

Fig. 1. Generalized Inertial Proximal Deblurring. We provide a 3D surface plot of a given Point Spread Function (PSF) along with a color-coded visualization of the PSF profile (*top-left corner*). We show a close-up view of the CHAMELEON's head for the reference image and the blurred image with the corresponding given PSF (*bottom-left corner*). Then, we display a side-by-side comparison of the CHAMELEON image deblurred by Forward-Backward [Combettes and Wajs 2005; Liu et al. 2012], Douglas-Rachford [Douglas and Rachford 1956], Viscosity [Imnang 2013; Kitkuan et al. 2019], Richardson [Richardson 1910] and our solver (*middle*). Finally, we plot the evolution of the convergence rate of our solver along 20000 iterations against this baseline (*right*).

Visual signal deblurring is a challenging computational problem involving spatially invariant point spread functions, large blurring matrices and deconvolution. We formulate the visual content restoration process as an inverse convex minimization problem. We design a novel iterative multi-steps scheme incorporating an inertial term to approximate an element of the set of solutions of accretive inclusion problems. We generalize our solver for a large variety of inverse problems in imaging such as convex minimization, variational inequality and split feasibility problems. We compare the convergence rate and perceptual quality assessment with state-of-the-art algorithms on various visual input data. We demonstrate the effectiveness of our solver to deblur RGB images, HDR images, height fields, geometry images as well as motion caption data.

CCS Concepts: • **Computing methodologies** → **Image Processing**.

1 RELATED WORK

Computational Deblurring. Wang *et al.* [Wang and Tao 2014] for an overview of recent progress in image deblurring. Applications related to visual blur covers understanding, modeling, removing or synthesizing blur. A significant difficulty is to identify the contribution of the defocus blur component [Mosleh et al. 2015] and the motion blur component [Jin et al. 2019] into the perceived blur. Late-breaking works in dynamic scene deblurring using spatially variant recurrent neural networks [Nah et al. 2019]. Blind image deblurring restores images without explicitly prior knowledge of the blur function [Chen et al. 2019]. Blind deconvolution requires the estimation of unknown PSF parameters [Li 2019] and have been extended to visual motif [Hertz et al. 2019]. For blind deblurring, we extend the sub-pixel technique of Delbracio *et al.* [Delbracio et al. 2012] to estimate a PSF from a single image.

Iterative Deconvolution Optimization. Least-Squares methods have been practical numerical solutions in Computer Graphics [Pighin and Lewis 2007]. We notice a long-standing interest in linear least-squares methods for image deblurring since the pioneering work of Helstrom [Helstrom 1967] and they remain attractive approaches well-studied in recent research [Jose et al. 2017]. Generally, no exact and unique solution may exist due to the numerical noise related to inverse problems. The estimated solution is obtained by converging toward the unknown solution using a collection of iterations steps. However, existing iterative methods can only provide an approximated deblurred images [Nagy et al. 2004]. Given a stopping criterion., the key challenge is to design an iterative scheme leading to an optimal set of parameters with a low-residual error and a high converging rate. Efficient image optimization includes proximal algorithms [Heide et al. 2016].

Iterative Approximation Schemes. Image restoration is a practical discrete inverse problem [Hansen 2010] for which computational methods [Vogel 2002] and regularization [Engl et al. 2000] exists. Image deblurring admits least-squares solutions [Stanimirovic et al. 2015]. For large-scale systems, numerical solutions are obtained via an iterative scheme generating a sequence of improving approximate solutions converging toward a global minimizer starting from an initial guess. Several works have studied weak and strong convergence criteria [Suantai 2005], rate of convergence [Phuengrattana and Suantai 2011] and convergence acceleration [Polyak 1964] of

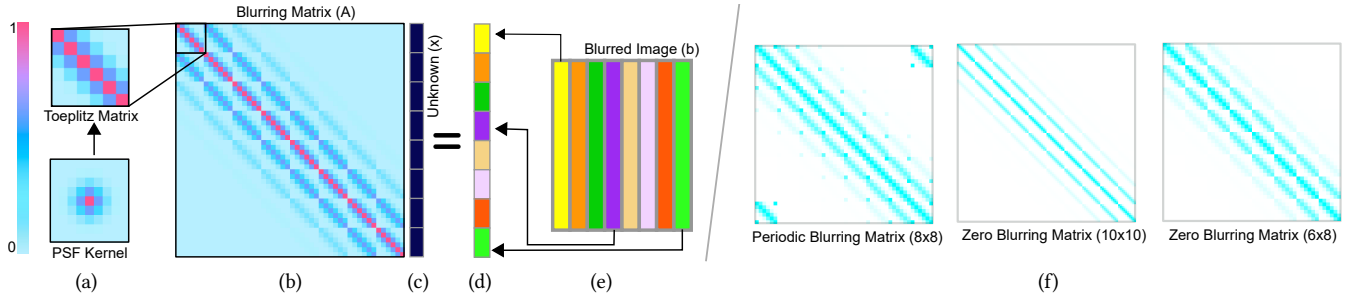Authors' addresses: Yann Savoye, ysavoye@siggraph.org, University of Leicester, United Kingdom; Damrongsak Yambangwai, damrongsak.ya@up.ac.th, University of Phayao, Thailand; Watcharaporn Cholamjiak, watcharaporn.ch@up.ac.th, University of Phayao, Thailand.

Fig. 2. Modeling the deblurring process. Spread Function (PSF). We show the Toeplitz matrix generated from the given PSF (**a**). The blurring matrix is a block Toeplitz matrix with Toeplitz blocks. Then, the blurring matrix A is block-based and each block corresponds to the Toeplitz matrix associated at a given entry of the blurring matrix (**b**). Next, we construct a linear least-squares system as follows: the left-hand side of the system stores the blurring matrix (**b**) and the right-hand side (**d**) is then flattened and column-wise blurred image (**b**). The estimated deblurred solution is the unknown (**c**). For the sake of simplicity, we did not show the Laplacian that could be stacked right under the left-hand side of the system. There are several choices for blurring matrix pattern, whatever is zero or periodic boundary condition. Finally, we display the color-coded matrix sparsity pattern of the periodic and zero blurring matrices for several sizes of an input image (**f**).

new iterative approximation schemes. Our proposed iterative methods are inspired by the work of Kitkuan *et al.* [Kitkuan et al. 2019] in which algorithms for zeros of two accretive operators are used to solve convex minimization problems in image restoration. In our work, we relax the fixed-point problem using iterative least-squares methods. Fixed-point theorems are generally proven in the Hilbert spaces [Takahashi 2010] or the Banach spaces [Browder 1965], but we restrict the proof of convergence and the resulting algorithm in the Euclidean space in which the image restoration problem lies. The viscosity iterative method [Imnang 2013], the Douglas method [Douglas and Rachford 1956] and the Richardson method [Richardson 1910] are competitive methods to our solver.

Proximal and Splitting Methods. Proximal splitting methods have been extensively used in signal processing [Combettes and Pesquet 2011] to improve robustness to numerical noise or image denoising [Heide et al. 2016]. Rockafellar [Rockafellar 1976] introduced a class of algorithms called proximal algorithms using the proximal operators of the objective terms to solve non-smooth, constrained, large-scale convex optimization problems. We refer the reader to the work of Parikh *et al.* [Parikh and Boyd 2014] for a comprehensive overview of proximal algorithms. Proximal gradient methods are forward-backward splitting methods used to solve possibly non-differentiable optimization problems [Tseng 2000]. Forward-backward splitting methods are generalized in Banach spaces by Cholamjiak [Cholamjiak 2016]. Inertial accelerated algorithms have been developed to accelerate the convergence rate [Dang et al. 2017]. The inertial term can be injected into the forward-backward method [Lorenz and Pock 2015], proximal method [Alvarez and Attouch 2001] and the splitting proximal method [Moudafi and Oliny 2003]. Finally, the split feasibility problem (SFP) captures a wide range of inverse problems. We combine the proximal, forward-backward, and inertial principles to outperform classical methods.

Image Quality Assessment. Image quality assessment [Lavoué and Mantiuk 2015] requires metrics for measuring or predicting the perceptual difference between two images [Wolski et al. 2018]. A large collection of image quality assessment have been developed from error visibility to local/structural similarity [Wang et al. 2004]. For instance, Wang *et al.* [Wang and Bovik 2002] has designed a universal image quality index (UIQ) modeling image distortion by mixing three quantities: loss of correlation, luminance distortion, and contrast distortion. Also, Xue *et al.* [Xue et al. 2014] proposed a perceptual image quality Index based on gradient magnitude similarity deviation. Specific visual metrics have also been proposed to assess dynamic range image quality [Aydin et al. 2008]. For instance, the HDR-VDP-2 method [Mantiuk et al. 2011] is one of those metrics in the dynamic range space. Moreover, weaknesses of existing image quality metrics in evaluating graphics artifacts are known [Cadik et al. 2012]. In the configuration in which no ground truth available, it is even more critical to assess image quality and thus, a no-reference metric is preferred for evaluating the quality of deblurring [Liu et al. 2013]. Recently, deep learning has been an interesting alternative for blind image quality assessment [Bianco et al. 2018].

2 DEBLURRING PROBLEM AND NUMERICAL SOLUTION

In this section, we provide technical background work related to the problem of blur modeling and image restoration as well as its optimization formulation with iterative numerical scheme.

Linear Blur Model. The Fredholm integral equation of the first kind which arises from many image or signal restoration problems is formulated as follows:

$$\int_a^b \kappa(s,t) x(t) \, dt = h(s) \tag{1}$$

where $\kappa(s,t)$ is integral kernel. The blurred function $h(s)$ is obtained by the known $\kappa(s,t)$ and the true function $x(t)$. We can derive a linear system by discretizing the integral Equation (1). Spatially variant image blur is modeled as the convolution of latent sharp image with a shift-invariant kernel plus noise η, which is typically considered to be additive white Gaussian noise. The degradation that we aim to recover form is modeled by:

$$h = \mathcal{K} \otimes x + \eta$$

with the original image x, the blurry degraded image h, a blur kernel $k \in \mathbb{R}^l$, the additive white Gaussian noise η with unknown standard deviation and $\mathcal{K}$ the PSF of the blurring operator. We denote by $\otimes$ the correlation operator of convolution.
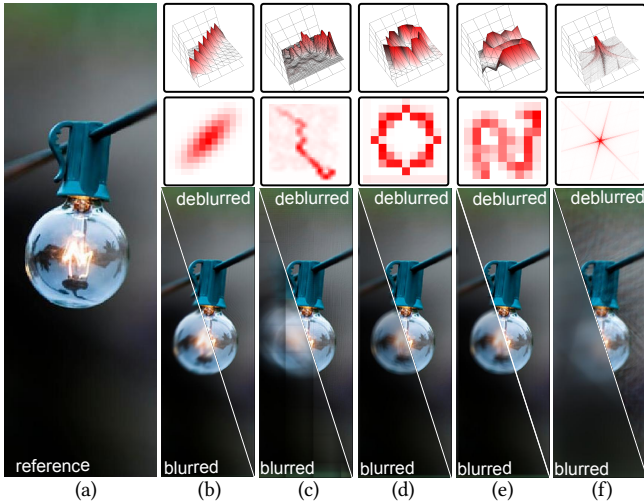
Fig. 3. Effects of Spatial PSF. We demonstrate the robustness of our solver under the various amount of blurriness. We display the reference image of the *bubble light* (**a**). We show the blurring/deblurring effect of our solver for different typical blur distribution patterns: linear motion blur (**b**), shake motion blur (**c**), circular motion blur (**d**), non-linear motion blur (**e**), glare for a 6-blade iris (**f**) with corresponding intensity distribution. These PSFs are empirical point spread function and have ideal symmetrical shapes, expressing the amount of blurring as determined by the wavelengths of light. The intensity distribution of PSFs are shown as 3D surface plots and color-coded profiles.

Image Restoration Problem. Generally, the deblurring problem is represented as a convex minimization problem. In particular, the idea is to solve the iterative deconvolution minimizing the following estimation error:

$$\arg\min_{\mathcal{I}} \left(\|\mathcal{B} - (\mathcal{I} \otimes \mathcal{K} + \eta)\|_2^2 + \lambda \|\nabla\mathcal{I}\|_2^2 \right)$$

with $\mathcal{K}$ the PSF of the blurring operator, η the additive white Gaussian noise with unknown standard deviation, $\otimes$ the correlation operator of convolution and $\lambda = e^{-3}$. We denote by $\mathcal{B} \in \mathbb{R}^{w \times h \times d}$ the observed image of size $N = w \times h$ pixels and d the dimension of the color space. We relate the given blurred image $\mathcal{B}$ to the unknown sharp image $\mathcal{I} \in \mathbb{R}^{w \times h \times d}$ with d channels by $\mathcal{B} = \mathcal{I} \otimes \mathcal{K} + \eta$ where $\mathcal{B}$ is defined as the ground truth image $\mathcal{I}$ blurred by a PSF $\mathcal{K}$ plus inevitable image noise distribution $\eta = \mathcal{N}(0, \sigma^2)$. The operator $\|\cdot\|_2$ is ℓ_2-norm.

Least-Squares Form. Given the stacked vector $\boldsymbol{c}$ from blurred image, and the blurring $\boldsymbol{W}$, the goal is to find an approximation of the stacked vector $\boldsymbol{x}$ of true image. We model the blurring of images as a linear process (see Figure 2). In our work, we cast the image restoration problem as a general unconstrained linear system such that:

$$\|\boldsymbol{Ax} - \boldsymbol{b}\|_2^2 = \|\boldsymbol{Wx} - \boldsymbol{c}\|_2^2 + \lambda \|\nabla\boldsymbol{x}\|_2^2$$
$$\Rightarrow \min_{\boldsymbol{x} \in \mathbb{R}} \left\| \begin{bmatrix} \boldsymbol{W} \\ \lambda\boldsymbol{L} \end{bmatrix} \boldsymbol{x} - \begin{bmatrix} \boldsymbol{c} \\ 0 \end{bmatrix} \right\|^2$$

where $\boldsymbol{x} \in \mathbb{R}^N$ is a column vector stacking the columns of the unknown original noise-free image $\mathcal{I}$ and a scalar $\lambda > 0$. Also, the stacked vector $\boldsymbol{c} \in \mathbb{R}^N$ is a column vector stacking the columns

of the observed blurring image. $\boldsymbol{W} \in \mathbb{R}^{N \times N}$ is the blurring matrix. Here, the discrete Laplacian $\boldsymbol{L}$ operator is an optional regularization term and can be seen as a PSF.

Toeplitz Blurring Matrix. The blurring matrix is bounded linear operator defined from point spread function. The blurring matrix $\boldsymbol{W} \in \mathbb{R}^{nm \times nm}$ with n is the number of rows in the image and m is the number of column in the image. The matrix $\boldsymbol{W}$ is very large and sparse matrix and composed of $m \times m$ sub-blocks. In 2D, the blurring matrix is a two-level Toeplitz matrix, namely a block Toeplitz matrix with Toeplitz blocks. Each sub-block of $\boldsymbol{W}$ is typically of size $n \times n$ and consists in a Toeplitz matrix. Using periodic boundary condition, the Toeplitz sub-block starting at the top-left corner (k, l) entry of $\boldsymbol{W}$ is denoted by $\boldsymbol{W}_{kl} \in \mathbb{R}^{n \times n}$ and defined by:

$$\left[\boldsymbol{W}_{k,l}\right]_{p,q} = \mathcal{K}\left[(p - q) \mod n, (k - l) \mod m\right].$$

Splitting Method. The reason $\boldsymbol{A}^{-1}\boldsymbol{b}$ cannot be used to deblur images is the amplification of high-frequency components of the noise in the data, caused by the inversion of very small singular values of $\boldsymbol{A}$. However, the solution $\boldsymbol{x}$ can be estimated by many iterations. Our minimization optimization can be written as the splitting of two convex functions f and g, as follows:

$$\min_{\boldsymbol{x} \in \mathbb{H}} (f(\boldsymbol{x}) + g(\boldsymbol{x})) \text{ with } f(\boldsymbol{x}) = \|\boldsymbol{Wx} - \boldsymbol{c}\|_2^2 \text{ and } g(\boldsymbol{x}) = \lambda \|\nabla\boldsymbol{x}\|_2^2$$

where $f, g : \mathbb{H} \to \mathbb{R}$ are two proper and lower semi-continuous convex functions such that f is differentiable with L-Lipschitz continuous gradient [Combettes and Pesquet 2011].
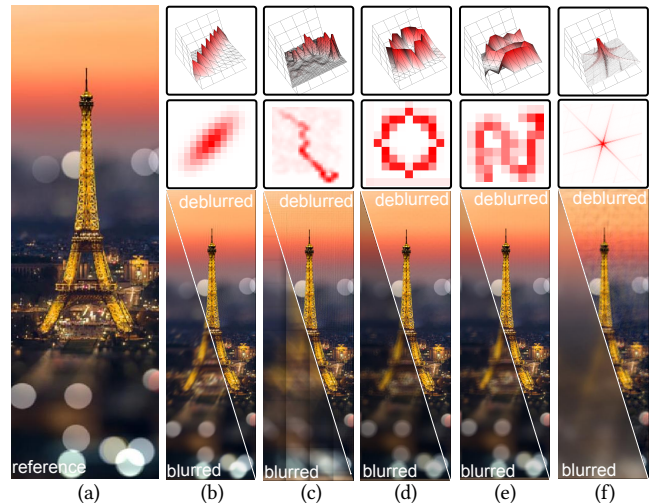

Fig. 4. Effects of Spatial PSF. We demonstrate the robustness of our solver under the various amount of blurriness. We display the reference image of the *Eiffel tower* (**a**). We show the blurring/deblurring effect of our solver for different typical blur distribution patterns: linear motion blur (**b**), shake motion blur (**c**), circular motion blur (**d**), non-linear motion blur (**e**), glare for a 6-blade iris (**f**).

Convex Minimization Problem. Let $\mathbb{H}$ be a Hilbert space and the functions $f : \mathbb{H} \to \mathbb{R}$ be a convex smooth function and $g : \mathbb{H} \to \mathbb{R}$ be a convex, lower-semicontinuous and nonsmooth function in $\mathbb{H}$.
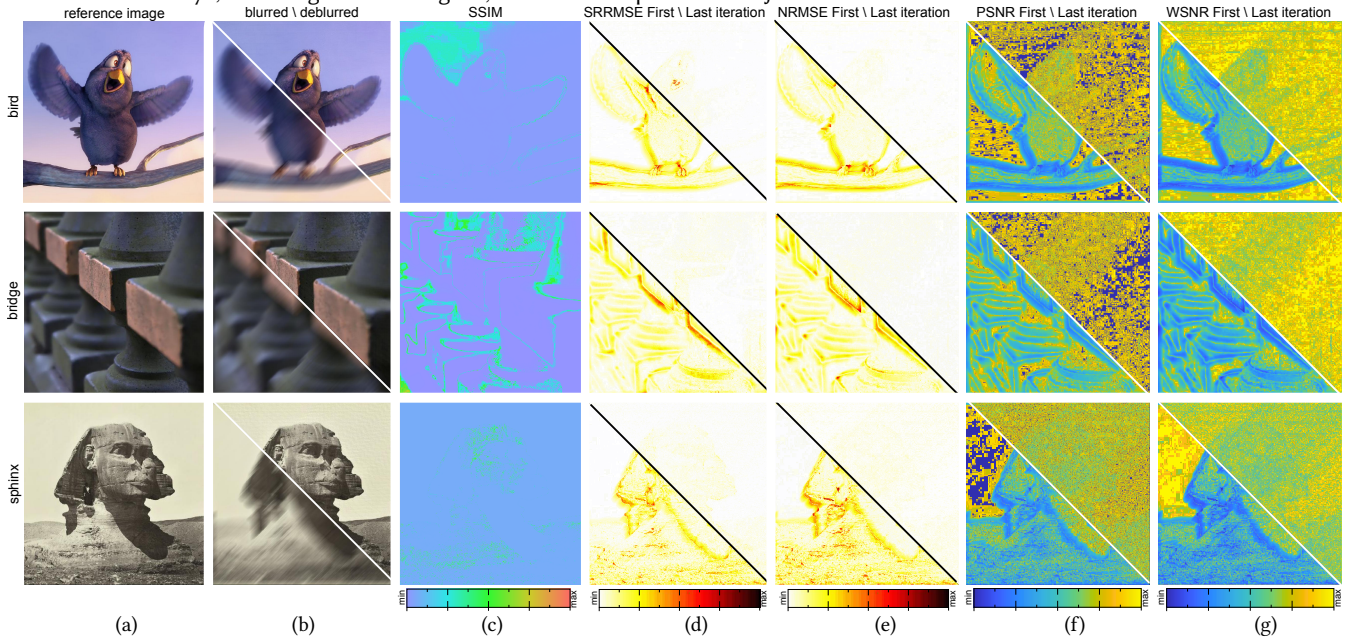
Fig. 5. Image Quality Assessment. We perform a quality and distortion image assessment of our deblurring solver on publicly available datasets such as the *bird* (*top row*), *bridge* (*middle row*) and *sphinx* (*bottom row*) images. First, we display reference images used as the ground truth image (**a**), as well as the blurred/deblurred images (**b**). Blurred images are obtained by corrupting the reference images with a motion blur of size length 21. Next, we show the color-coded SSIM (**c**) between the reference and deblurred images. Finally, we display a side-by-side comparison of the color-coded visualization for the SRRMSE (**d**), NRMSE (**e**), PSNR (**f**) and WSNR (**g**) at the first and last iteration of our deblurring solver with respect to the reference image.

We consider the problem of finding a point x^* in the Hilbert space, such that for all $x \in \mathbb{H}$,

$$f\left(x^*\right) + g\left(x^*\right) \le f\left(x\right) + g\left(x\right).$$

In particular, we define $F = \nabla f$ an operator and $G = \partial g$ a multi-valued operator, with ∇f is the gradient of f. The subdifferential operator is noted ∂f. We denote by ∂g the subdifferential of g defined by:

$$\partial g\left(x\right) := \left\{s \in \mathbb{H} : g\left(y\right) \ge g\left(x\right) + \langle s, y - x\rangle, \ \forall y \in \mathbb{H}\right\}.$$

The operator T can be written as the sum of two maximal monotone operators, i.e. $T = F + G$ from $\mathbb{H}$ to $2^{\mathbb{H}}$ such that

$$J_{\lambda(F+G)} = (F + G)^{-1}(0) = \{z \in \mathbb{H} \mid 0 \in (Fz + Gz)\} \ne \emptyset.$$

The set of zero point $T^{-1}(0)$ defined by $T^{-1}(0) = \{z : 0 \in Tz\}$. Then, $T^{-1}(0)$ equal to the set fixed point of $J_{\lambda T}$. The problem is to iteratively find a zero of the sum of two monotone operators F and G in a Hilbert space $\mathbb{H}$ is equivalent to find $x^* \in \mathbb{H}$ a solution to the inclusion problem such that:

$$0 \in \nabla f\left(x^*\right) + \partial g\left(x^*\right) \text{ also written as } 0 \in (F + G)\left(x^*\right).$$

The ∇g can be used if g is smooth (i.e. real-valued and differentiable everywhere with Lipschitzian gradient), otherwise g is used proximally via its proximity operator. For any $\lambda \in (0, +\infty)$, its solutions are characterized by the fixed point equation

$$x = \text{prox}_{\lambda g}\left(x - \lambda \nabla f\left(x\right)\right).$$

The proximal operator $\text{prox}_{\lambda g}$ of the scaled function λg, with $\lambda > 0$, reduced to Euclidean projection onto $\mathbb{C}$ a closed non-empty convex set is expressed as:

$$\text{prox}_{\lambda g}\left(v\right) = \underset{x \in \mathbb{C}}{\text{argmin}}\left(g\left(x\right) + \frac{1}{2\lambda}\left\|x - v\right\|_2^2\right)$$

where $\|\cdot\|$ is the Euclidean norm. The previous equation suggests the possibility of iterating

$$x_{n+1} = \underbrace{\text{prox}_{\lambda_n g}}_{\text{backward step}} \underbrace{\left(x_n - \lambda_n \nabla f\left(x_n\right)\right)}_{\text{forward step}}.$$

The mapping $J_{\lambda \partial g} = (I + \lambda \partial g)^{-1}$ is called the resolvent of operator ∂g with parameter $\lambda > 0$ and I is the identity, in such way that the proximal operator is the resolvent of the subdifferential operator. The fixed points of the proximal operator of g are the minimizers of g. We assume that x^* is a solution of g if and only if $J_{\lambda \partial g}\left(x^*\right) = x^*$. Then, $\text{prox}_{\lambda f}\left(x^*\right) = x^*$ if and only if x^* minimizes g. We obtain the the following equality:

$$\text{prox}_{\lambda g} = (I + \lambda \partial g)^{-1}.$$

The forward-backward scheme uses the recursive application of an explicit forward step with respect to F, followed by an implicit backward step with respect to G. The forward-backward splitting method is defined by:

$$x_0 \in \mathbb{H} \text{ and } x_{n+1} = (I + \lambda G)^{-1}\left(x_n - \lambda F x_n\right), \ n \ge 1.$$

In the equations, the subscript refers to the nth iteration. Also, we consider the splitting iterative methods [Lions and Mercier 1979] in a real Hilbert space at the $(n + 1)^{th}$ iteration:

$$x_{n+1} = \left(2J_\lambda^F - I\right)\left(2J_\lambda^G - I\right)x_n, \ n \ge 1 \text{ and}$$

$$x_{n+1} = J_\lambda^F\left(2J_\lambda^G - I\right)x_n + \left(I - J_\lambda^G\right)x_n, \ n \ge 1,$$
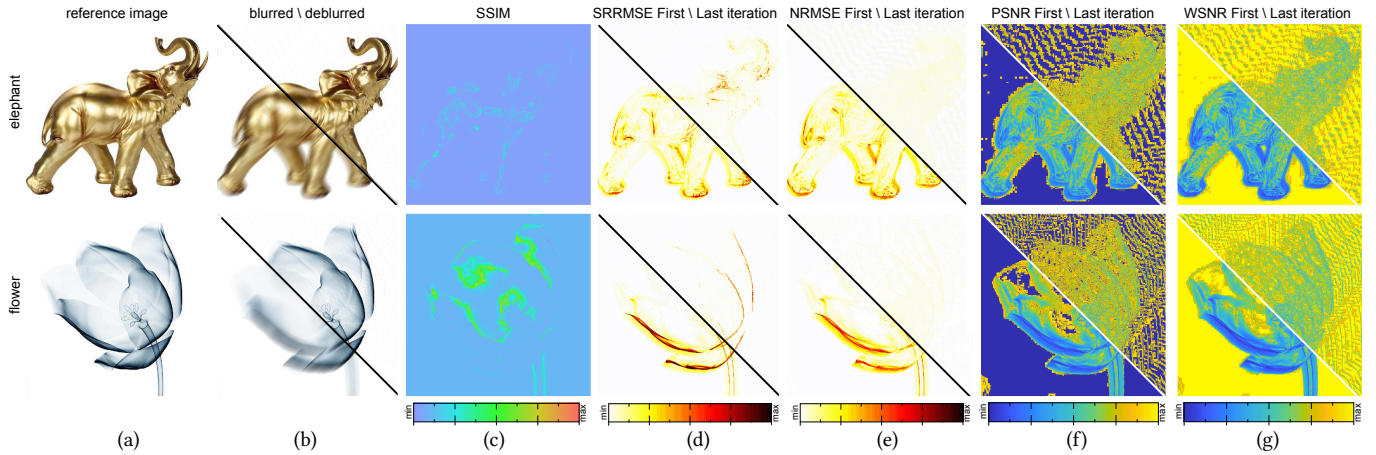
Fig. 6. Image Quality Assessment. We perform quality and distortion image assessment of our deblurring solver for the *elephant* (**top row**) and *flower* (**bottom row**) images. First, we display reference images used as the ground truth image (**a**), as well as the blurred/deblurred images (**b**). Blurred images are obtained by corrupting the reference images with a motion blur of size length 21. Next, we show the color-coded *Single Scale Structural Similarity* (SSIM) (**c**) between the reference and deblurred images. Finally, we display a side-by-side comparison of the color-coded visualization for the *Standardized Residual Root Mean Square Error* (SRRMSE) (**d**), the *Normalized Root Mean Square Error* (NRMSE) (**e**), the *Peak Signal to Noise Ratio* (PSNR) (**f**) and *Weighted Signal to Noise Ratio* (WSNR) (**g**) at the first and last iteration of our deblurring solver with respect to the reference image.

where $J_\lambda^F = (I + \lambda F)^{-1}$ and where $J_\lambda^G = (I + \lambda G)^{-1}$ with $\lambda > 0$. The proximity operator of g is denoted $\text{prox}_{\lambda_n g} (I + \lambda \partial g)^{-1}$ with the step size $\lambda > 0$. Then, the problem becomes

$$x_{n+1} = \text{prox}_{\lambda g} (x_n - \lambda \nabla f (x_n)), n \geq 1.$$

Next, we consider the inertial proximal point algorithm converges weakly to a zero of G with the following form:

$$\begin{cases} y_n = x_n + \theta_n (x_n - x_{n-1}) \\ x_{n+1} = (I + \lambda_n G)^{-1} y_n, \ n \geq 1. \end{cases}$$

If $\{\lambda_n\}$ is non-decreasing and $\{\theta_n\} \subset [0, 1)$ with

$$\sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\|^2 < \infty.$$

$\{\alpha_n\}_{n \in \mathbb{N}}$ is a sequence in $[0, 2]$ so that $\sum_{n \in \mathbb{N}} \alpha_n (2 - \alpha_n) = \infty$.

Let $\tau \in (0, \infty)$, and $x_0 \in \mathbb{H}$. Set

$$\begin{cases} u_n = J_\tau^G x_n \\ y_n = J_\tau^F (2u_n - x_n) \\ x_{n+1} = x_n + \alpha_n (y_n - u_n). \end{cases}$$

3 OUR NUMERICAL SOLVER

We provide the theorem for our new iterations with the inertial technical term and a forward-backward method for solving the constrained problem as follows:

$$\min_{x \in \mathbb{R}^N} \frac{1}{2} \|Ax - b\|_2^2 + g(x) \qquad (2)$$

where $A : \mathbb{R}^N \to \mathbb{R}^M$ is bounded linear operator, $b \in \mathbb{R}^M$ is fixed and $g : \mathbb{R}^N \to (-\infty, \infty]$ is proper, lower-semicontinuous and convex function.

Proposed Theorem. Let $A : \mathbb{R}^N \to \mathbb{R}^M$ be a bounded linear operator and $b \in \mathbb{R}^M$ with K the largest eigenvalue of $A^T A$. Let $\{x_n\}$ be a sequence generated by $x_0, x_1 \in \mathbb{R}^N$ and

$$\begin{cases} y_n &= x_n + \theta_n (x_n - x_{n-1}), \\ z_n &= (1 - \gamma_n) y_n + \gamma_n \, \text{prox}_{\tau_n g} \left(y_n - \tau_n A^T (A y_n - b) \right), \\ u_n &= (1 - \beta_n - \delta_n) z_n + \delta_n y_n \\ &\quad + \beta_n \, \text{prox}_{\tau_n g} \left(z_n - \tau_n A^T (A z_n - b) \right), \\ x_{n+1} &= (1 - \alpha_n - \sigma_n) u_n + \sigma_n z_n \\ &\quad + \alpha_n \, \text{prox}_{\tau_n g} \left(u_n - \tau_n A^T (A u_n - b) \right), \end{cases}$$

where $\{\tau_n\} \subset (0, 2/K)$, $\{\theta_n\} \subset [0, \theta]$ for some $\theta \in [0, 1)$ and $\{\alpha_n\}, \{\beta_n\}, \{\gamma_n\}, \{\delta_n\}, \{\sigma_n\}$ are sequences in $[0, 1]$. To establish theoretical convergence, we need to assume that suitable conditions hold. The suitable conditions to be assumed are the following:

$$(i) \quad \sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty,$$

$$(ii) \quad 0 < \liminf_{n \to \infty} \gamma_n < \limsup_{n \to \infty} \gamma_n < 1,$$

$$(iii) \quad \limsup_{n \to \infty} \delta_n < 1, \limsup_{n \to \infty} \sigma_n < 1,$$

$$(iv) \quad 0 < \liminf_{n \to \infty} \tau_n \leq \limsup_{n \to \infty} \tau_n < \frac{2}{K}.$$

Then, $\{x_n\}$ converges to a minimizer of the problem (described in Equation (2)).

Remark. We remark here that the condition *(i)* is easy to implement since the value of $\|x_n - x_{n-1}\|$ is known before choosing θ_n. Indeed, the parameter θ_n can be chosen such that $0 \leq \theta_n \leq \bar{\theta}_n$, where

$$\bar{\theta}_n = \begin{cases} \min \left\{ \frac{\omega_n}{\|x_n - x_{n-1}\|}, \theta \right\} & \text{if } x_n \neq x_{n-1} \\ \theta & \text{otherwise} \end{cases}$$

where $\{\omega_n\}$ is a positive sequence such that $\sum_{i=n}^{\infty} \omega_n < \infty$. We use the following parameters:

$$\tau_n = \frac{1}{\|A\|_1 \|A\|_\infty}, \ \theta = \frac{1}{4}, \ \alpha_n = \frac{n}{100n + 1}$$

and $\alpha_n = \beta_n = \gamma_n$. Then, $\{x_n\}$ converges strongly.

4 SOLVER PSEUDO-CODE

The algorithm described previously in our proposed theorem is easy and straightforward to implement as detailed in the corresponding pseudo-code (see Algorithm 1).

Algorithm 1 Generalized Inertial Proximal Deblurring

Input: A, left-hand side containing the blurring matrix
Input: b, right-hand side containing blurred signal.
Input: c, maximum number of iterations.
Output: x, unknown restored signal.

1: **function** SOLVER(A, b)
2:    /* Initialization */
3:    $n \leftarrow 1$
4:    /* vector of maximum values of A column-wise */
5:    $x_0 \leftarrow \max_{\text{col}}(A)$
6:    /* vector of minimum values of A column-wise */
7:    $x_1 \leftarrow \min_{\text{col}}(A)$
8:    /* Iterations */
9:    **while** $(n < c)$ and $(\|x_n - x_{n-1}\| < 10^{-5})$ **do**
10:      /* Parameters Update */
11:      $\tau_n \leftarrow \frac{1}{\|A\|_1 \|A\|_\infty}$
12:      $\alpha_n \leftarrow 1 - \frac{n}{100n+1}$
13:      $\beta_n \leftarrow \gamma_n = \alpha_n$
14:      $\sigma_n \leftarrow \delta_n = \frac{1}{100n^2+1}$
15:      /* Step 1 - find θ_n */
16:      $\theta_n = \begin{cases} \min\left(\frac{1}{n^2\|x_n - x_{n-1}\|}, 0.25\right) & \text{if } x_n \neq x_{n-1} \\ 0.25 & \text{otherwise} \end{cases}$
17:      /* Step 2 - find y_n */
18:      $y_n \leftarrow x_n + \theta_n (x_n - x_{n-1})$
19:      /* Step 3 - find z_n */
20:      $z_n \leftarrow (1 - \gamma_n) y_n$
          $+ \gamma_n \, \text{prox}_{\tau_n g}\left(y_n - \tau_n A^T (A y_n - b)\right)$
21:      /* Step 4 - find u_n */
22:      $u_n \leftarrow (1 - \beta_n - \delta_n) z_n + \delta_n y_n$
          $+ \beta_n \, \text{prox}_{\tau_n g}\left(z_n - \tau_n A^T (A z_n - b)\right)$
23:      /* Step 5 - find x_{n+1} */
24:      $x_{n+1} \leftarrow (1 - \alpha_n - \sigma_n) u_n + \sigma_n z_n$
          $+ \alpha_n \, \text{prox}_{\tau_n g}\left(u_n - \tau_n A^T (A u_n - b)\right)$
25:      $n \leftarrow n + 1$
26:    **return** x

The parameters used in our pseudo-code are the extrapolation parameter (θ_n), the convergence step size (τ_n), the SP-parameters (α_n, β_n) and the generalized SP parameters (δ_n, σ_n). The convergence step size τ remains constant for all iterations (*line 11*). To ensure convergence, the convergence step size must satisfy $0 < \tau < \frac{2}{\sigma_{max}^2}$ with σ_{max} is the largest singular valued of A. The inertial term is represented by the term $\theta_n (x_n - x_{n-1})$. This inertial term is related to the condition $\sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty$. We detail the iterative steps of our algorithm as follows. The first step (*lines 15-16*) computes θ_n as the extrapolation parameter for the current iteration. The second step (*line 17*) computes the inertial term y_n using the

heavy Ball method [Polyak 1964, 1987] written as $\theta_n (x_n - x_{n-1})$. Then, from the third step to fifth step, we generalize SP-iterations. The third step (*line 19*) performs the forward backward step z_n. The fourth step (*line 19*) performs the Richardson iteration u_n. The Richardson iteration is generally used as an iterative regularization method. The fifth step (*lines 23-24*) returns the overall result for the current iteration. We note that $\{\alpha_n\}$, $\{\beta_n\}$ and $\{\gamma_n\}$ are sequences in $[0, 1]$. We choose x_0 and x_1 to be respectively the vector stacking the minimum and maximum image intensity per-column of the input image (*lines 4-7*). Finally, the stopping criterion is defined by $\|x_{n+1} - x_n\| < 10^{-5}$ (*line 9*). The stopping criterion is set to the residual norm $\|b - Ax_n\| < 10^{-5}$ to ensure algorithmic convergence regardless of the nature of the data. We provide a proof of convergence.

5 ALGORITHM PROOF

We provide a rigorous proof for the theorem of convergence and its resulting algorithm mentioned above. We split this proof into five comprehensive steps.

• *First Step:* In this first step, we show that $\forall x, y \in \mathbb{R}^N$, $\tau \in (0, 2/K)$, we have the follow inequality:

$$\left\| \text{prox}_{\tau g}\left(x - \tau A^T (Ax - b)\right) - \text{prox}_{\tau g}\left(y - \tau A^T (Ay - b)\right) \right\| \leq \|x - y\|$$

Since A is bounded linear operator, we have the following expression:

$$\left\| \tau A^T (Ax - b) - \tau A^T (Ay - b) \right\|^2$$
$$= \left\| \tau A^T A (x - y) \right\|^2$$
$$= \langle \tau A^T A (x - y), \ \tau A^T A (x - y) \rangle$$
$$\leq \tau K \langle x - y, \ \tau A^T A (x - y) \rangle.$$

This derivation implies that

$$\langle x - y, \tau A^T A (x - y) \rangle \geq \frac{1}{\tau K} \left\| \tau A^T (Ax - b) - \tau A^T (Ay - b) \right\|^2 .$$

It follows from Equation 3 and $\tau \in (0, 2/K)$ that

$$\left\| \text{prox}_{\tau g}\left(x - \tau A^T (Ax - b)\right) - \text{prox}_{\tau g}\left(y - \tau A^T (Ay - b)\right) \right\|^2$$

$$\leq \left\| x - \tau A^T (Ax - b) - \left(y - \tau A^T (Ay - b)\right) \right\|^2$$
$$= \left\| (x - y) - \tau A^T A (x - y) \right\|^2$$
$$= \|x - y\|^2 - \left\| \tau A^T A (x - y) \right\|^2$$
$$\quad - 2\langle (x - y) - \tau A^T A (x - y), \tau A^T A (x - y) \rangle$$
$$= \|x - y\|^2 + \left\| \tau A^T A (x - y) \right\|^2$$
$$\quad - 2\langle (x - y), \tau A^T A (x - y) \rangle$$
$$\leq \|x - y\|^2 + \left(1 - \frac{2}{rK}\right) \left\| \tau A^T A (x - y) \right\|^2 \leq \|x - y\|^2$$

		Error Metrics			Quality Metrics					
		RMSLE	NRMSE	SRRMSE	SSIM	DSSIM	UQI	SNR	WSNR	PSNR
Bird	Forward-Backward	0.01435 ± 1.6	0.16554 ± 2.8	0.02530 ± 1.8	0.03388 ± 0.7	0.48342 ± 0.3	0.02214 ± 44.7	14.25029 ± 53.8	−34.13791 ± 55.8	39.1146
	Douglas-Rachford	0.01065 ± 1.6	0.11947 ± 2.6	0.01889 ± 1.8	0.03511 ± 0.9	0.48280 ± 0.4	0.03561 ± 30.2	14.23081 ± 49.6	−27.89751 ± 60.2	42.2228
	Viscosity	0.01440 ± 1.6	0.16593 ± 2.8	0.02539 ± 1.8	0.03387 ± 0.7	0.48343 ± 0.3	0.02217 ± 45.4	14.24476 ± 54.0	−34.19123 ± 55.7	39.0783
	Richardson	0.01435 ± 1.6	0.16555 ± 2.8	0.02530 ± 1.8	0.03388 ± 0.7	0.48342 ± 0.3	0.02211 ± 44.0	14.25023 ± 53.7	−34.13960 ± 55.8	39.1147
	Ours	0.00891 ± 1.5	0.09796 ± 2.4	0.01586 ± 1.8	0.03555 ± 0.8	0.48257 ± 0.4	0.05659 ± 38.8	14.23118 ± 52.3	−24.06918 ± 62.9	**44.1166**
Bridge	Forward-Backward	0.01201 ± 0.8	0.07820 ± 1.0	0.02129 ± 1.7	0.07312 ± 0.9	0.46405 ± 0.4	0.03869 ± 7.0	13.10795 ± 33.2	−19.23074 ± 49.6	46.6249
	Douglas-Rachford	0.01086 ± 0.8	0.07078 ± 0.8	0.01925 ± 1.3	0.07344 ± 0.9	0.46389 ± 0.4	0.04370 ± 6.9	13.10684 ± 33.1	−17.78394 ± 48.9	47.8850
	Viscosity	0.01199 ± 0.8	0.07829 ± 1.0	0.02126 ± 1.7	0.07311 ± 0.9	0.46405 ± 0.4	0.03877 ± 7.0	13.10783 ± 33.1	−19.21875 ± 49.7	46.6022
	Richardson	0.01201 ± 0.8	0.07820 ± 1.0	0.02129 ± 1.7	0.07312 ± 0.9	0.46405 ± 0.4	0.03869 ± 7.0	13.10795 ± 33.2	−19.23074 ± 49.9	46.6249
	Ours	0.01058 ± 0.6	0.06898 ± 0.8	0.01876 ± 1.2	0.07359 ± 0.9	0.46381 ± 0.4	0.04466 ± 6.9	13.10303 ± 33.0	−17.51636 ± 49.2	**48.4250**
Sphinx	Forward-Backward	0.02039 ± 1.1	0.25287 ± 2.5	0.03571 ± 1.6	0.01614 ± 0.9	0.49232 ± 0.4	−0.02448 ± 214.6	10.40382 ± 52.6	−34.69215 ± 48.8	35.1107
	Douglas-Rachford	0.01813 ± 1.0	0.22495 ± 1.9	0.03176 ± 1.3	0.01679 ± 0.9	0.49200 ± 0.4	−0.03774 ± 233.0	10.40546 ± 53.1	−33.11753 ± 45.9	36.4943
	Viscosity	0.02045 ± 1.1	0.25338 ± 2.5	0.03581 ± 1.6	0.01613 ± 0.9	0.49233 ± 0.4	−0.02511 ± 204.4	10.40269 ± 52.2	−34.70973 ± 49.1	35.0897
	Richardson	0.02039 ± 1.1	0.25287 ± 2.5	0.03571 ± 1.6	0.01614 ± 0.9	0.49232 ± 0.4	−0.02449 ± 214.7	10.40382 ± 52.6	−34.69227 ± 48.8	35.1107
	Ours	0.01701 ± 0.9	0.21120 ± 1.9	0.02979 ± 1.3	0.01711 ± 0.9	0.49184 ± 0.4	−0.03330 ± 226.2	10.40383 ± 52.9	−32.22014 ± 47.3	**37.2093**

Table 1. Quantitative Evaluation and Comparison: We perform quantitative evaluation of image quality assessment metrics for our solver on the *bird*, *bridge* and *sphinx* images. We provide measurement between deblurred and reference images for RMSLE, NRMSE, SRRMSE, SSIM, DSSIM, UQI, SNR, WSNR, PSNR. We compare statistics of our solver with Forward-Backward [Combettes and Wajs 2005; Liu et al. 2012], Douglas-Rachford [Douglas and Rachford 1956], Viscosity [Imnang 2013; Kitkuan et al. 2019], Richardson [Richardson 1910].

		Error Metrics			Quality Metrics					
		RMSLE	NRMSE	SRRMSE	SSIM	DSSIM	UQI	SNR	WSNR	PSNR
Elephant	Forward-Backward	0.01270 ± 1.3	0.12996 ± 3.4	0.02250 ± 1.6	0.00798 ± 0.9	0.49642 ± 0.4	0.00125 ± 8.0	13.45384 ± 52.8	−21.46062 ± 65.2	39.1037
	Douglas-Rachford	0.00995 ± 1.4	0.09925 ± 3.2	0.01782 ± 1.7	0.00826 ± 0.9	0.49629 ± 0.4	0.00972 ± 103.3	13.51303 ± 73.6	−18.03994 ± 67.3	41.9135
	Viscosity	0.01273 ± 1.3	0.13039 ± 3.3	0.02255 ± 1.6	0.00798 ± 0.9	0.49643 ± 0.4	0.00113 ± 8.0	13.45396 ± 52.8	−21.49621 ± 64.5	39.0765
	Richardson	0.01270 ± 1.3	0.12996 ± 3.4	0.02250 ± 1.6	0.00798 ± 0.9	0.49642 ± 0.4	0.00125 ± 8.0	13.45384 ± 52.8	−21.46062 ± 65.2	39.1037
	Ours	0.00844 ± 1.2	0.08152 ± 3.1	0.01526 ± 1.6	0.00840 ± 0.9	0.49621 ± 0.4	0.00272 ± 4.7	13.46599 ± 47.0	−15.56681 ± 69.1	**43.9694**
Flower	Forward-Backward	0.01277 ± 2.2	0.13509 ± 3.7	0.02208 ± 1.9	0.04682 ± 0.9	0.47771 ± 0.4	0.00348 ± 6.8	22.46092 ± 37.3	−24.80973 ± 70.2	36.1787
	Douglas-Rachford	0.01099 ± 1.9	0.11903 ± 3.0	0.01962 ± 1.9	0.04769 ± 0.9	0.04769 ± 0.9	0.47728 ± 0.4	0.00350 ± 7.9	−23.76566 ± 67.9	37.7950
	Viscosity	0.01280 ± 2.2	0.13534 ± 3.7	0.02210 ± 1.9	0.04677 ± 0.9	0.47774 ± 0.4	0.00332 ± 7.0	22.46047 ± 37.3	−24.82603 ± 70.3	36.1553
	Richardson	0.01277 ± 2.2	0.13509 ± 3.7	0.02208 ± 1.9	0.04682 ± 0.9	0.47771 ± 0.4	0.00348 ± 6.8	22.46092 ± 37.3	−24.80973 ± 70.2	36.1787
	Ours	0.01006 ± 1.9	0.11150 ± 2.7	0.01826 ± 1.9	0.04816 ± 0.9	0.47705 ± 0.4	0.00276 ± 9.3	22.44692 ± 35.4	−23.24203 ± 66.2	**38.6136**

Table 2. Quantitative Evaluation and Comparison: We perform a image quality assessment for our solver on the *elephant* and *flower* images. We provide measurement between deblurred and reference images for the *Root Mean Squared Log Error* (RMSLE), *Normalized Root Mean Square Error* (NRMSE), *Standardized Residual Root Mean Square Error* (SRRMSE), *Single Scale Structural Similarity* (SSIM), *Structural Dissimilarity* (DSSIM), *Universal Quality Index* (UQI), *Signal-to-Noise Ratio* (SNR), *Weighted Signal-to-Noise Ratio* (WSNR), *Peak Signal-to-Noise Ratio* (PSNR). We compare statistic of our solver with Forward-Backward [Combettes and Wajs 2005; Liu et al. 2012], Douglas-Rachford [Douglas and Rachford 1956], Viscosity [Imnang 2013; Kitkuan et al. 2019], Richardson [Richardson 1910]. Our method outperforms state-of-the-art proximal methods by producing the most competitive performance.

• *Second Step:* We exhibit in this second step that $\{x_n\}$ is bounded. Let p denotes the following expression $\arg\min\left\{\frac{1}{2}\|Ax - b\|_2^2 + g(x)\right\}$ and $S_\tau x = \operatorname{prox}_{\tau g}\left(x - \tau A^T(Ax - b)\right)$. From the first step, we derive the following inequalities:

$$
\begin{aligned}
\|x_{n+1} - p\| &\leq (1 - \alpha_n - \sigma_n)\|u_n - p\| + \sigma_n\|z_n - p\| \\
&\quad + \alpha_n\left\|S_{\tau_n} u_n - p\right\| \\
&\leq (1 - \sigma_n)\|u_n - p\| + \sigma_n\|z_n - p\| \\
&\leq (1 - \sigma_n)\left((1 - \beta_n - \delta_n)\|z_n - p\| \right. \\
&\quad \left. + \delta_n\|y_n - p\| + \beta_n\left\|S_{\tau_n} z_n - p\right\|\right) + \sigma_n\|z_n - p\| \\
&\leq (1 - \sigma_n)\left((1 - \delta_n)\|z_n - p\| \right. \\
&\quad \left. + \delta_n\|y_n - p\|\right) + \sigma_n\|z_n - p\| \\
&\leq (1 - \sigma_n)\left((1 - \delta_n)\left((1 - \gamma_n)\|y_n - p\| \right.\right. \\
&\quad \left.\left. + \gamma_n\left\|S_{\tau_n} y_n - p\right\|\right) + \delta_n\|y_n - p\|\right) \\
&\quad + \sigma_n\left((1 - \gamma_n)\|y_n - p\| + \gamma_n\left\|S_{\tau_n} y_n - p\right\|\right) \\
&\leq \|y_n - p\| \\
&\leq \|x_n - p\| + \theta_n\|x_n - x_{n-1}\|.
\end{aligned}
$$

From Alvarez's Lemma [Alvarez and Attouch 2001] and the assumption *(i)*, $\lim_{n\to\infty}\|x_n - p\|$ exists. In particular, $\{x_n\}$, $\{y_n\}$, $\{z_n\}$ and $\{u_n\}$ are bounded.

• *Third Step:* We highlight that $\lim_{n\to\infty}\left\|S_{\tau_n} y_n - y_n\right\| = 0$. From the first step, we derive the following inequalities:

$$
\begin{aligned}
\|x_{n+1} - p\|^2 &\leq (1 - \alpha_n - \sigma_n)\|u_n - p\|^2 \\
&\quad + \sigma_n\|z_n - p\|^2 + \alpha_n\left\|S_{\tau_n} y_n - p\right\|^2 \\
&\leq (1 - \sigma_n)\|u_n - p\|^2 + \sigma_n\|z_n - p\|^2 \\
&\leq (1 - \sigma_n)\left((1 - \beta_n - \delta_n)\|z_n - p\|^2 \right. \\
&\quad \left. + \delta_n\|y_n - p\|^2 + \beta_n\left\|S_{\tau_n} z_n - p\right\|^2\right) \\
&\quad + \sigma_n\|z_n - p\|^2 \\
&\leq (1 - \sigma_n)\left((1 - \delta_n)\|z_n - p\|^2 \right. \\
&\quad \left. + \delta_n\|y_n - p\|^2\right) + \sigma_n\|z_n - p\|^2 \\
&= \left((1 - \sigma_n)(1 - \delta_n) + \sigma_n\right)\|z_n - p\|^2 \\
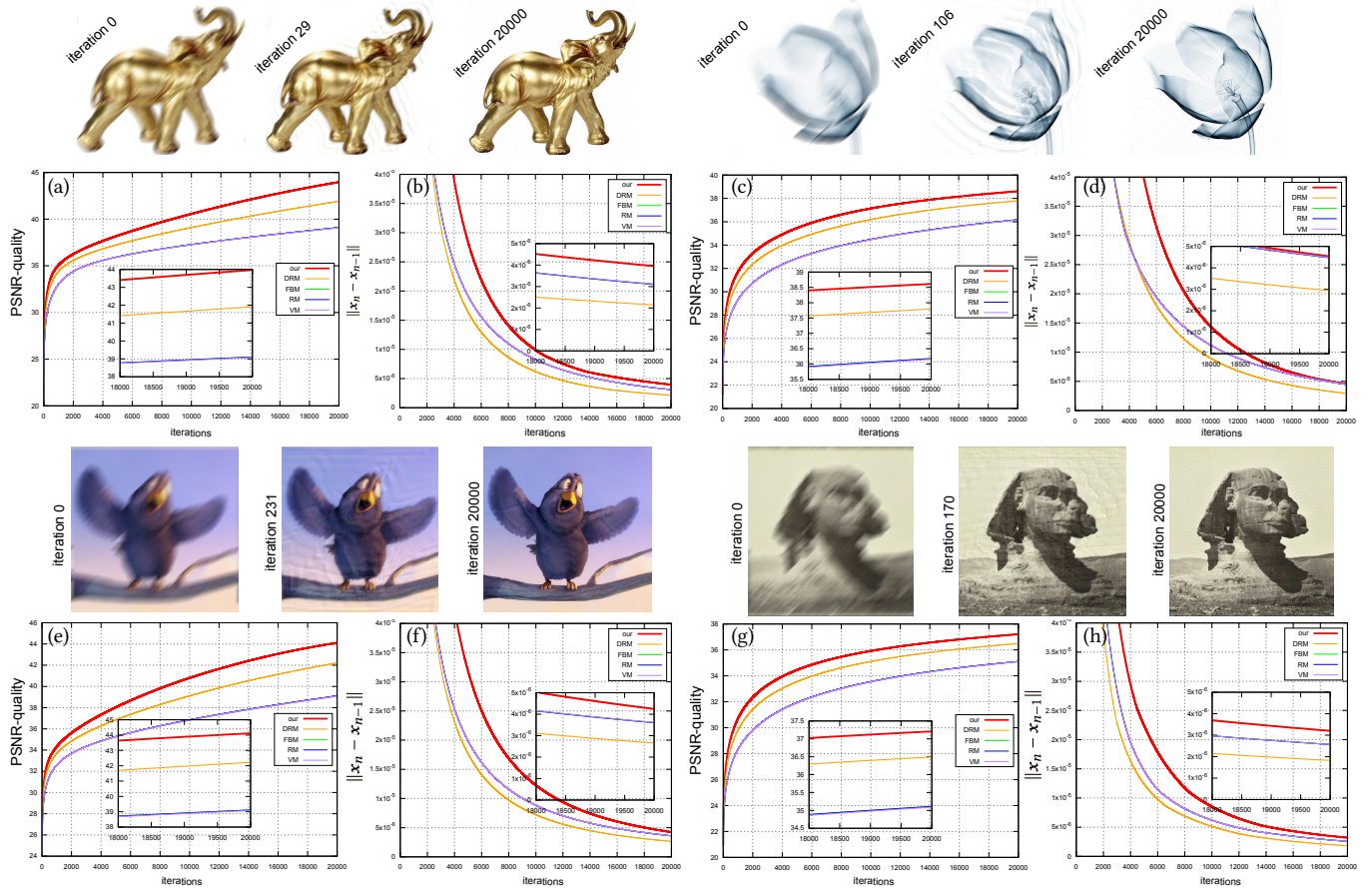&\quad + (1 - \sigma_n)\delta_n\|y_n - p\|^2
\end{aligned}
$$

Fig. 7. Convergence Rate and Quality Evolution. We compare the PNSR quality and convergence error for our solver against Forward-Backward (FBM) [Combettes and Wajs 2005; Liu et al. 2012], Douglas-Rachford (DRM) [Douglas and Rachford 1956], Viscosity (VM) [Imnang 2013; Kitkuan et al. 2019] and Richardson (RM) [Richardson 1910]. We plot the PSNR quality (**a,c,e,g**) and convergence error $\|x_{n+1} - x_n\|$ (**b,d,f,h**) respectively for the *elephant* (**top left**), *flower* (**top right**), *bird* (**bottom left**), *sphinx* (**bottom right**) images. We provide intra-iteration visual quality of the deblurring over the estimated image to cross-reference qualitative with quantitative evaluation. The convergence of our algorithm to the optimal solution is guaranteed by the convexity of the cost function used for the adaptation. We depict the performance of our solver in red color. Our solver demonstrates constant superiority over the baseline with fastest convergence and fewer iterations than the standard methods.

$$
\begin{aligned}
\|x_{n+1} - p\|^2 &= ((1 - \sigma_n)(1 - \delta_n) + \sigma_n) \\
&\quad \left((1 - \gamma_n) \|y_n - p\|^2 + \gamma_n \|S_{\tau_n} y_n - p\|^2 \right. \\
&\quad \left. - (1 - \gamma_n) \gamma_n \|S_{\tau_n} y_n - y_n\|^2 \right) \\
&\quad + (1 - \sigma_n) \delta_n \|y_n - p\|^2 \\
&\leq \|y_n - p\|^2 - ((1 - \sigma_n)(1 - \delta_n) + \sigma_n) \\
&\quad (1 - \gamma_n) \gamma_n \|S_{\tau_n} y_n - y_n\|^2 \\
&\leq \|x_n - p\|^2 + 2\theta_n \langle x_n - x_{n-1}, y_n - p \rangle \\
&\quad - ((1 - \sigma_n)(1 - \delta_n) + \sigma_n) \\
&\quad (1 - \gamma_n) \gamma_n \|S_{\tau_n} y_n - y_n\|^2 \\
&\leq \|x_n - p\|^2 + 2\theta_n \|x_n - x_{n-1}\| \|y_n - p\| \\
&\quad - ((1 - \sigma_n)(1 - \delta_n) + \sigma_n)(1 - \gamma_n) \\
&\quad \gamma_n \|S_{\tau_n} y_n - y_n\|^2 .
\end{aligned}
$$

This derivation implies that

$$
\begin{aligned}
&((1 - \sigma_n)(1 - \delta_n) + \sigma_n)(1 - \gamma_n) \gamma_n \|S_{\tau_n} y_n - y_n\|^2 \\
&\leq \|x_n - p\|^2 - \|x_{n+1} - p\|^2 \\
&\quad + 2\theta_n \|x_n - x_{n-1}\| \|y_n - p\| . \tag{3}
\end{aligned}
$$

Given that $\lim\limits_{n \to \infty} \|x_n - p\|$ exists and taking the assumptions *(i)-(iii)*, we can state from Equation (3) that $\lim\limits_{n \to \infty} \|S_{\tau_n} y_n - y_n\| = 0$.

• *Fourth Step:* We demonstrate that $\{x_n\}$ converges to some point $q \in \mathbb{R}^N$. Since $\{y_n\}$ is bounded, it exists a sub-sequence $\{y_{n_k}\}$ of $\{y_n\}$ converging to $q_1 \in \mathbb{R}^N$. We take another sub-sequence $\{y_{n_j}\}$ of $\{y_n\}$ converging to $q_2 \in \mathbb{R}^N$. Assuming $q_1 \neq q_2$, then by the Opial's condition of $\mathbb{R}^N$, we derive the following expression:

$$
\begin{aligned}
\liminf_{n \to \infty} \|y_n - q_1\| &= \liminf_{k \to \infty} \|y_{n_k} - q_1\| \\
&< \liminf_{k \to \infty} \|y_{n_k} - q_2\| \\
&= \liminf_{n \to \infty} \|y_n - q_2\|
\end{aligned}
$$

$$
\begin{aligned}
&= \liminf_{j \to \infty} \left\| \boldsymbol{y}_{n_j} - q_2 \right\| \\
&< \liminf_{j \to \infty} \left\| \boldsymbol{y}_{n_j} - q_1 \right\| \\
&= \liminf_{n \to \infty} \left\| \boldsymbol{y}_n - q_1 \right\|
\end{aligned}
$$

which is a contradiction.

Therefore, $q_1 = q_2$. We show that $\{\boldsymbol{y}_n\}$ converges to $q \in \mathbb{R}^N$. By definition of $\{\boldsymbol{x}_n\}$ and our assumption (i), we get the following equality:

$$
\lim_{n \to \infty} \|\boldsymbol{x}_n - \boldsymbol{y}_n\| = \lim_{n \to \infty} \theta_n \|\boldsymbol{x}_n - \boldsymbol{x}_{n-1}\| = 0.
$$

This equality implies that $\{\boldsymbol{x}_n\}$ also converges to $q \in \mathbb{R}^N$.
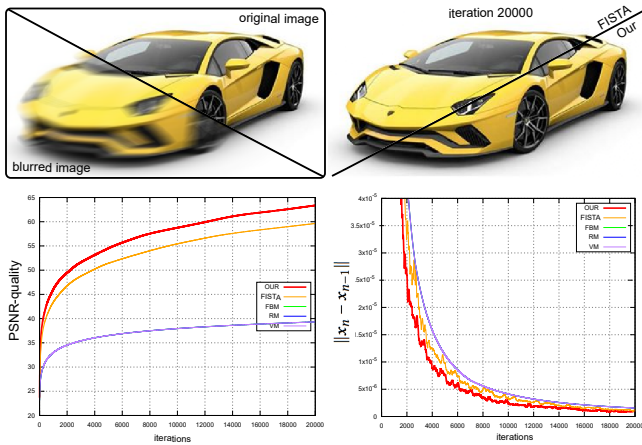

Fig. 8. Comparison with FISTA. We provide the reference image of the *Lamborghini* and the image corrupted with a motion blur of size length 21 (*top left*). Then, we display a side-by-side comparison of the deblurred image using the Fast proximal gradient methods (FISTA) [Beck and Teboulle 2009] and our solver after 20000 iterations (*top right*). Our solver obtains better PSNR quality (*bottom right*) and faster convergence (*bottom right*) than the FISTA.

• *Fifth Step:* In this next step, we would like to show that $q = \arg\min \left\{ \frac{1}{2} \|\boldsymbol{Ax} - \boldsymbol{b}\|_2^2 + g(\boldsymbol{x}) \right\}$. First, we show that for any $\varepsilon > 0$, there exists a positive number $\varphi(\varepsilon) > 0$ such that $\|S_\tau \boldsymbol{x} - \boldsymbol{x}\| < \varepsilon$ for all $\boldsymbol{x} \in co(\{\boldsymbol{x}_0, \boldsymbol{x}_1\})$, $r \in \left(0, \frac{2}{K} \right)$, whenever $\boldsymbol{x}_0, \boldsymbol{x}_1 \in \mathbb{R}^N$ with $\|S_\tau \boldsymbol{x}_0 - \boldsymbol{x}_0\| \leq \varphi(\varepsilon)$ and $\|S_\tau \boldsymbol{x}_1 - \boldsymbol{x}_1\| \leq \varphi(\varepsilon)$. Let $\boldsymbol{x} = (1 - \lambda)\boldsymbol{x}_0 + \lambda \boldsymbol{x}_1$, for some $\lambda \in [0, 1]$ and $\varepsilon > 0$. We consider the following two cases.

Case I. If $\|\boldsymbol{x}_0 - \boldsymbol{x}_1\| < \frac{\varepsilon}{3}$, then

$$
\|\boldsymbol{x} - \boldsymbol{x}_0\| = \lambda \|\boldsymbol{x}_0 - \boldsymbol{x}_1\| < \frac{\varepsilon}{3}.
$$

If $\varphi(\varepsilon) < \frac{\varepsilon}{3}$, then we obtain the following inequality from the first step:

$$
\begin{aligned}
\|S_\tau \boldsymbol{x} - \boldsymbol{x}\| &\leq \|S_\tau \boldsymbol{x} - S_\tau \boldsymbol{x}_0\| \\
&\quad + \|S_\tau \boldsymbol{x}_0 - \boldsymbol{x}_0\| + \|\boldsymbol{x}_0 - \boldsymbol{x}\| \\
&\leq 2 \|\boldsymbol{x}_0 - \boldsymbol{x}\| + \|S_\tau \boldsymbol{x}_0 - \boldsymbol{x}_0\| \\
&< 2 \left(\frac{\varepsilon}{3} \right) + \varphi(\varepsilon) < \varepsilon.
\end{aligned}
$$

Case II. If $\|\boldsymbol{x}_0 - \boldsymbol{x}_1\| \geq \frac{\varepsilon}{3}$, then for any non-negative number λ such that $\lambda < \frac{\varepsilon}{3\|\boldsymbol{x}_0 - \boldsymbol{x}_1\|}$, we have

$$
\|\boldsymbol{x} - \boldsymbol{x}_0\| = \lambda \|\boldsymbol{x}_0 - \boldsymbol{x}_1\| < \frac{\varepsilon}{3}.
$$

If $\varphi(\varepsilon) < \frac{\varepsilon}{3}$ and $\lambda < \frac{\varepsilon}{3\|\boldsymbol{x}_0 - \boldsymbol{x}_1\|}$, we derive the following inequality from the first step:

$$
\begin{aligned}
\|S_\tau \boldsymbol{x} - \boldsymbol{x}\| &\leq \|S_\tau \boldsymbol{x} - S_\tau \boldsymbol{x}_0\| + \|S_\tau \boldsymbol{x}_0 - \boldsymbol{x}_0\| \\
&\quad + \|\boldsymbol{x}_0 - \boldsymbol{x}\| \\
&\leq 2 \|\boldsymbol{x}_0 - \boldsymbol{x}\| + \|S_\tau \boldsymbol{x}_0 - \boldsymbol{x}_0\| \\
&< 2 \left(\frac{\varepsilon}{3} \right) + \varphi(\varepsilon) < \varepsilon.
\end{aligned}
$$

We assume that $\lambda \in \left[\frac{\varepsilon}{3\|\boldsymbol{x}_0 - \boldsymbol{x}_1\|}, 1 \right]$ and $\|\boldsymbol{x}_0 - \boldsymbol{x}_1\| \geq \frac{\varepsilon}{3}$.

From the first step, we obtain the following equality:

$$
\begin{aligned}
\|S_\tau \boldsymbol{x} - \boldsymbol{x}_0\| &\leq \|S_\tau \boldsymbol{x} - S_\tau \boldsymbol{x}_0\| + \|S_\tau \boldsymbol{x}_0 - \boldsymbol{x}_0\| \\
&\leq \|\boldsymbol{x} - \boldsymbol{x}_0\| + \varphi(\varepsilon) \\
&= \lambda \|\boldsymbol{x}_1 - \boldsymbol{x}_0\| + \varphi(\varepsilon) \quad (4)
\end{aligned}
$$

as well as the next following equality:

$$
\begin{aligned}
\|S_\tau \boldsymbol{x} - \boldsymbol{x}_1\| &\leq \|S_\tau \boldsymbol{x} - S_\tau \boldsymbol{x}_1\| + \|S_\tau \boldsymbol{x}_1 - \boldsymbol{x}_1\| \\
&\leq \|\boldsymbol{x} - \boldsymbol{x}_1\| + \varphi(\varepsilon) \\
&= (1 - \lambda) \|\boldsymbol{x}_1 - \boldsymbol{x}_0\| + \varphi(\varepsilon). \quad (5)
\end{aligned}
$$

From Equations (4) and (5) and $\lambda \in \left[\frac{\varepsilon}{3\|\boldsymbol{x}_0 - \boldsymbol{x}_1\|}, 1 \right]$, we obtain

$$
\begin{aligned}
\|S_\tau \boldsymbol{x} - \boldsymbol{x}\| &\leq (1 - \lambda) \|S_\tau \boldsymbol{x} - \boldsymbol{x}_0\| + \lambda \|S_\tau \boldsymbol{x} - \boldsymbol{x}_1\| \\
&\leq 2(1 - \lambda) \lambda \|\boldsymbol{x}_1 - \boldsymbol{x}_0\| + \varphi(\varepsilon) < \varepsilon.
\end{aligned}
$$

From the third step, we know that $\lim_{n \to \infty} \left\| S_{\tau_n} \boldsymbol{y}_n - \boldsymbol{y}_n \right\| = 0$. We set $\varepsilon_n = \left\| S_{\tau_n} \boldsymbol{y}_n - \boldsymbol{y}_n \right\|$. Let $\varepsilon > 0$. Since $\varepsilon_n \to 0$ as $n \to \infty$, there exists $M \in \mathbb{N}$ such that

$$
\varepsilon_n < \varepsilon \quad , \quad \forall n \geq M.
$$

We proof that $\boldsymbol{z} \in \bar{co}(\{\boldsymbol{y}_n : n \geq M\})$, $\left\| S_{\tau_n} \boldsymbol{z} - \boldsymbol{z} \right\| < \varepsilon$. We denote by $\bar{co}$ the closed convex hull. By the compactness of $\bar{co}(\{\boldsymbol{y}_n : n \geq M\})$, containing the limit q of $\{\boldsymbol{y}_n\}$. Next, $\left\| S_{\tau_n} \boldsymbol{q} - \boldsymbol{q} \right\| < \varepsilon$, $\forall n \geq M$. Hence $\left\| S_{\tau_n} \boldsymbol{q} - \boldsymbol{q} \right\| = 0$ and $S_{\tau_n} \boldsymbol{q} = \boldsymbol{b}$ since ε is arbitrary. Finally, we obtain $q = \text{prox}_{\tau_n g} \left(\boldsymbol{q} - \tau_n \boldsymbol{A}^T (\boldsymbol{Aq} - \boldsymbol{b}) \right)$, resulting in the expression of $q = \arg\min \left\{ \frac{1}{2} \|\boldsymbol{Ax} - \boldsymbol{b}\|_2^2 + g(\boldsymbol{x}) \right\}$.

6 DEBLURRING APPLICATIONS AND RESULTS

We have implemented a simple framework for PSF-based image deblurring both in *Matlab* and C++. We have designed a PSF editor and established a library of PSF to evaluate our solver. We demonstrate the effectiveness and practicability of our solver for several visual computing applications using RGB images (with red, green,
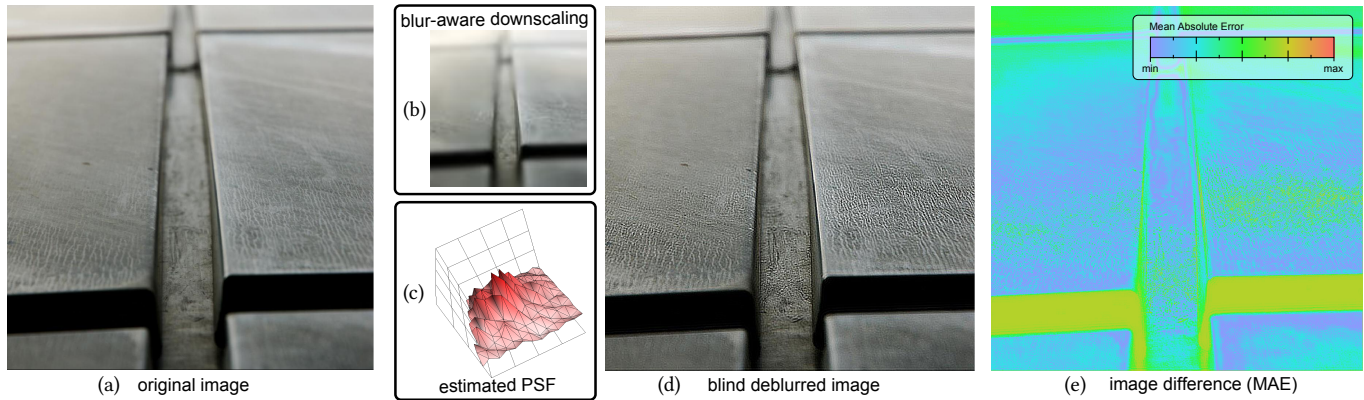
(a) original image blur-aware downscaling (b) (c) estimated PSF (d) blind deblurred image (e) image difference (MAE)

Fig. 9. Blind Deblurring. Our solver is also capable of performing blind deconvolution to output a blind deblurred image (**d**) from a single input image (**a**) without prior assumption of a ground truth PSF. Prior to the deblurring process, we estimate a non-parametric PSF from a single photography in two steps. We generate a low-resolution image using blur-aware image down-sampling [Trentacoste et al. 2011] and then we estimate the unknown point spread function using the sub-pixel estimation method [Delbracio et al. 2012] (**b**,**c**). Finally, we use our solver to approximate a sharp image (**d**) with the estimated PSF. Since no-reference is existing, The *Mean Absolute Error* (MAE) between the original input image and the blind deblurred images (**e**) shows that our solver preserves desirable depth-of-field backgrounds while it deblurs undesirable blurred foreground near high frequencies.

and blue color components for each individual pixel), height fields, high dynamic range images (HDR images) and skeletal motion data. Finally, we show results for blind deconvolution and the blurring effects on geometry images.

RGB, HDR, Grayscale Images. We solve the color deblurring problem using our algorithm. We show several restored examples of RGB images corrupted by a motion blur (see Figures 3 and 4). In addition, we deblur height fields in the grayscale space and visualize the impact of the deblurring on 3D surface. As seen in Figure 11, we demonstrate the usefulness for terrain systems. Finally, we developed an application to filter image in the high dynamic range [Chaurasiya and Ramakrishnan 2013]. We perform deblurring in the conventional RGBE space and we show high dynamic range tone mapping on deblurred HDR images in Figure 12 for various compression and saturation [Mantiuk et al. 2009].

Geometric and Motion Data. We have developed an application to blur/deblur in geometry image space [Gu et al. 2002]. In Figure 10, we render 3D bunny shapes from blurred and deblurred geometry images using our solver. Our solver is able to undo strong global blurring geometric shearing around the bunny head region. However, we observed that a deblurred geometry image leads to a noisy mesh. Generating a 3D mesh from a deblurred image reveals that the noise is perceptually amplified by the sensitiveness of geometry image meshing step to noise. Inherent to all numerical solvers, the deconvolution process increases noise, which exists at all frequencies in the image. To combat noise amplification prevent excessive artifacts during deblurring, we have enhanced our system with Graph Laplacian regularization to denoise by smoothing the error. Finally, we have developed an application to deblur skeletal motion data, as shown in Figure 13.

7 EVALUATION AND DISCUSSION

Effects of the PSF. We demonstrate our solver's robustness for non-Gaussian-like blurriness represented by highly non-uniform PSFs as seen in the real camera-captured *bubble light* images (see Figure 3) and *Eiffel Tower* images (see Figure 4). We show the effectiveness of our solver to deblur several typical blur distribution patterns such as linear, shake, circular and non-linear motion blurs. Our solver performs accurate deblurring even for images having strong glare blur or lens flare.

Quality Image Assessment. We perform extensive quality and distortion image assessment of our numerical solver for the *bird*, *bridge* and *sphinx* datasets (see Figure 5, Table 1) and for the *elephant* and *flower* datasets (see Figure 6, Table 2). Blurred images are obtained by corrupting the reference images with a motion blur of size length 21. We show the color-coded SSIM between the reference and deblurred images. Error function that measures the distance of the recovery estimate from the known image information. We use error functions to measure the distance of the recovery estimate from the known image information. The Structural Similarity Index Metric (SSIM) accounts for spatial information and is more consistent with human visual perception than traditional measures (such as PNSR or MSE). Finally, we provide color-coded visualization for the Standardized Residual Root Mean Square Error (SRRMSE), the Normalized Root Mean Square Error (NRMSE), the Peak Signal to Noise Ratio (PSNR) and Weighted Signal to Noise Ratio (WSNR) at the first and last iteration of our deblurring solver with respect to the reference image. In Figure 7, we also provide some intermediate visual results at different iterations of our solver.

Comparison to Baseline. Figure 1 shows a side-by-side qualitative and quantitative comparison for the *Chameleon* image deblurred by Forward-Backward [Combettes and Wajs 2005; Liu et al. 2012], Douglas-Rachford [Douglas and Rachford 1956], Viscosity [Imnang 2013; Kitkuan et al. 2019], Richardson [Richardson 1910] and owe
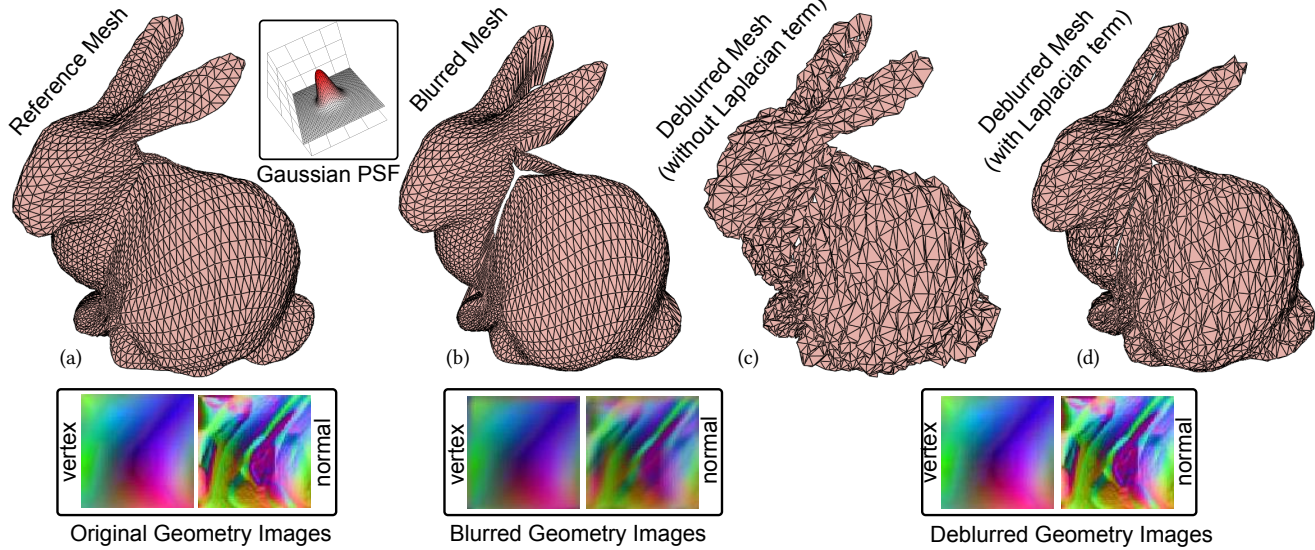
Fig. 10. Deblurring in Geometry Image Space. Geometry image captures geometry as a simple vertices array of 12-bit $[x, y, z]$ visualized as $[r, g, b]$ values and normal-map image of 8-bit $[n_x, n_y, n_z]$ visualized as $[r, g, b]$. In our example, we render 3D *bunny* shapes from original geometry images (**a**), blurred geometry images (**b**) using a simple Gaussian PSF and deblurred geometry images using our solver (**c**). We observe the so-called noise amplification inherently injected by any deconvolution process. This amplification is coupled with the sensitivity to noise of geometry image meshing (**c**). Finally, we combat the amplification of noise and prevent excessive artifacts during deblurring by applying Graph Laplacian regularization for denoising (**d**).

plot the evolution of the convergence rate along 20000 iterations of our solver against this baseline. In Table 1, we compare statistics of our solver with Forward-Backward [Combettes and Wajs 2005; Liu et al. 2012], Douglas-Rachford [Douglas and Rachford 1956], Viscosity [Imnang 2013; Kitkuan et al. 2019], Richardson [Richardson 1910]. We provide measurement between deblurred and reference images for Root Mean Squared Log Error (RMSLE), Normalized Root Mean Square Error (NRMSE), Standardized Residual Root Mean Square Error (SRRMSE), Single Scale Structural Similarity (SSIM), Structural Dissimilarity (DSSIM), Universal Quality Index (UQI), Signal-to-Noise Ratio (SNR), Weighted Signal-to-Noise Ratio (WSNR), Peak Signal-to-Noise Ratio (PSNR). Our method outperforms state-of-the-art proximal methods by producing the most competitive performance (essentially based on PSNR and RMSLE).
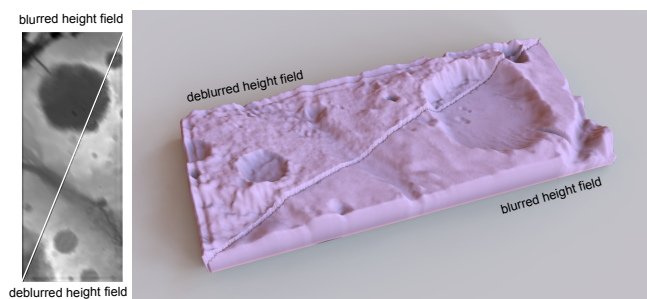

Fig. 11. Deblurring Height Field. We show a deblurring application in which a rasterized grayscale heightmap is blurred using Gaussian Point Spread Function and deblurred using our proposed solver in 2D space (*left*). We display side-by-side the corresponding generated blurred/deblurred 2.5D surface meshes of Mars to demonstrate the usefulness of our numerical solver to deblur maps for height-field–based terrain systems (*right*).

More importantly, the steps of alternating direction method of multipliers (ADMM) [Boyd et al. 2011] are mathematically equivalent to Douglas-Rachford used in our comparison in which our proposed iterations are performing better in terms of image quality. Our approach convergences faster while requiring fewer iterations. Moreover, our proofs had the same level of difficulty as other algorithms in which parameters are chosen in the real line interval (e.g., ADMM [Boyd et al. 2011]). Our method is robust to inaccurate PSFs as seen in the blind deconvolution of *floor tile* image (see Figure 9) because the estimated PSF is an approximation. The deconvolution of HDR images using RGBE representation shows that our solver is effective in floating-point continuous space. The geometry image is also an interesting application in which we can visualize and observe the noise amplification inherent to any blurring solver. This noise amplification on 3D surface can be reduced using Laplacian regularization or by a stopping converge criteria to prevent excessive artifacts.

Quality and Convergence Error. Finally, we plot in Figures 8 and 7 the PNSR quality and convergence error $\|x_{n+1} - x_n\|$ along with iterations for our solver against the same baseline and the same dataset to cross-reference qualitative with quantitative evaluation. Our solver demonstrates constant superiority over the state-of-the-art since the convergence is the fastest and it requires a smaller number of iterations than standard methods. The convergence of our solver is the fastest, and it requires a smaller number of iterations than standard methods, including the fast proximal gradient methods (FISTA) [Beck and Teboulle 2009] as shown in Figure 8 on the rendered *Lamborghini*. The convergence behavior of our method is guaranteed by the Cauchy error $\|x_{n+1} - x_n\|$. We have established theoretical convergence under some suitable conditions with detailed proof. Our proof demonstrates true convergence, which
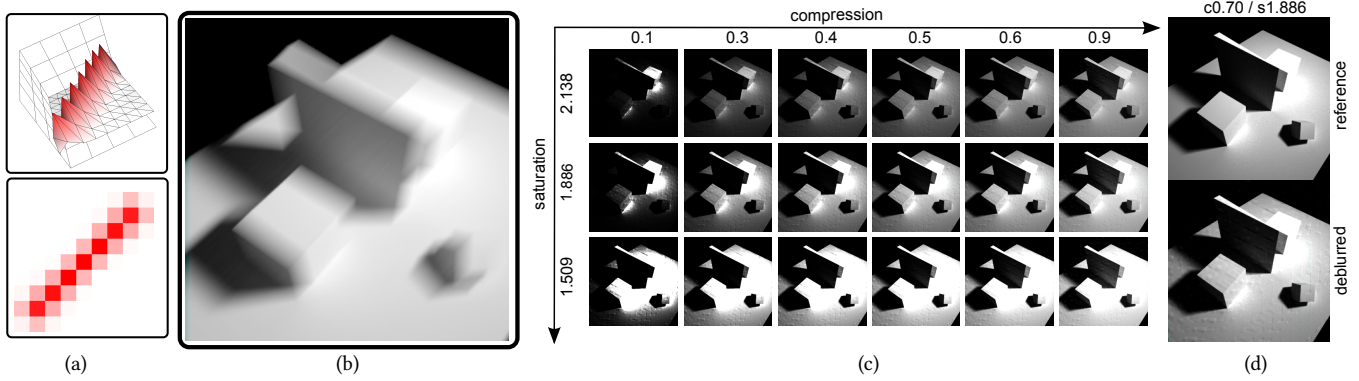
Fig. 12. Deblurring in the RGBE Space. We show a blurred HDR image with a motion blur (**a**). We provide a 3D surface plot of the corresponding motion blur PSF along with a color-coded visualization of its profile (**a**). Next, we perform deblurring in the RGBE space by applying our solver on each (R, G, B, E) floating-point channel. Then, we show the consistency of the resulting deblurred HDR by providing high dynamic range tone mapping results from the deblurred HDR image for various compression and saturation tone mapping (**c**). Finally, we show the reference and deblurred images side-by-side at compression $c = 0.70$ and saturation $s = 1.886$ (**d**).

guarantees the theorem and informs us about choosing of a suitable initialization for hyperparameters. The convergence to the optimal solution is guaranteed by the convexity of the cost function used for the adaptation.
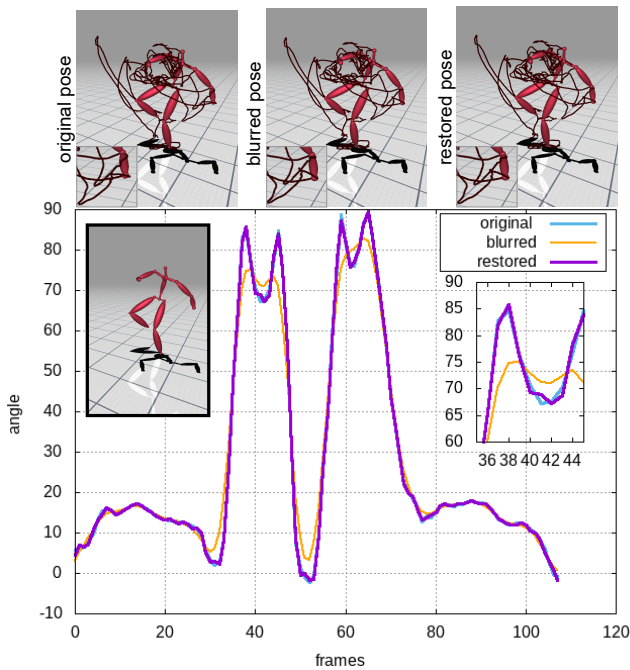

Fig. 13. Deblurring Skeletal Mocap. We deblur corrupted motion capture data using 1D Gaussian PSF with our solver. The deblur effect is shown on the motion curves. We also plot the reference, blurred and deblurred motion curve for a given joint in which the variation over time for the Euler angle is displayed as function of sampled keyframes.

Matrix-Vector Operation. We choose matrix-vector operation because it works better when the image is not large, then the matrix-vector operation method is more efficient since it does not require a

lot of conversion time. Generally, FFT works better when the image is large. The linear term is split from the proximal update for the convex regularizer and updates are computed for an inertial term (applied on an extrapolated estimation of the current iteration), forward-backward step and a Richardson regularization step.

8 CONCLUSIONS

Our work developed a generalized inertial proximal method, an iterative solver based on forward-backward iterations with a modified inertial term using proximal operators. This proximal optimization algorithm is designed to solve the convex deconvolution optimizations for visual deblurring applications. We generalized iterations to solve a large variety of constrained minimization problems in image restoration with wide applicability. The theoretical analysis proves the convergence property of our algorithm for solving constrained image restoration problems.

We provides both theoretical and experimental results to demonstrate the convergence properties of the proposed numerical algorithm. We demonstrated performance against standard forward-backward methods with an extensive experimental evaluation based on estimated or known PSFs. Our results show a substantial increase in PSNR quality compared to other deconvolution methods. Finally, we illustrated the generality of our numerical approach on a large spectrum of inputs such as geometry images, HDR image, height field and skeletal motion capture. At the frontier of pure mathematics and computer vision, we believe that our work represents a helpful technical tool with wide applicability beyond deblurring. For future work, we recommend the design of an approach to accelerate convergence using an automatic adjustment of the step size.

REFERENCES

Felipe Alvarez and Hedy Attouch. 2001. An Inertial Proximal Method for Maximal Monotone Operators via Discretization of a Nonlinear Oscillator with Damping. *Set-Valued Analysis* (2001).

Tunç Ozan Aydin, Rafal Mantiuk, Karol Myszkowski, and Hans-Peter Seidel. 2008. Dynamic Range Independent Image Quality Assessment. *ACM Trans. Graph.* (2008).

Amir Beck and Marc Teboulle. 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences* (2009).

Simone Bianco, Luigi Celona, Paolo Napoletano, and Raimondo Schettini. 2018. On the use of deep learning for blind image quality assessment. *Signal, Image and Video Processing* (2018).

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. 2011. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.* (2011).

Felix E. Browder. 1965. Nonexpansive Nonlinear Operators in a Banach Space. *Proceedings of the National Academy of Sciences* (1965).

Martin Cadik, Robert Herzog, Rafal Mantiuk, Karol Myszkowski, and Hans-Peter Seidel. 2012. New Measurements Reveal Weaknesses of Image Quality Metrics in Evaluating Graphics Artifacts. *ACM Trans. Graph.* (2012).

R. K. Chaurasiya and K. R. Ramakrishnan. 2013. High Dynamic Range Imaging. In *2013 International Conference on Communication Systems and Network Technologies.*

L. Chen, F. Fang, T. Wang, and G. Zhang. 2019. Blind Image Deblurring With Local Maximum Gradient Prior. In *CVPR.*

Prasit Cholamjiak. 2016. A Generalized Forward-backward Splitting Method for Solving Quasi Inclusion Problems in Banach Spaces. *Numer. Algorithms* (2016).

Patrick L. Combettes and Jean-Christophe Pesquet. 2011. Proximal Splitting Methods in Signal Processing. In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering.*

Patrick L. Combettes and Valérie R. Wajs. 2005. Signal Recovery by Proximal Forward-Backward Splitting. *Multiscale Modeling & Simulation* (2005).

Yazheng Dang, Jie Sun, and Honglei Xu. 2017. Inertial accelerated algorithms for solving a split feasibility problem. *Journal of Industrial & Management Optimization* (2017).

Mauricio. Delbracio, Andrés. Almansa, Jean-Michel. Morel, and Pablo. Musé. 2012. Subpixel Point Spread Function Estimation from Two Photographs at Different Distances. *SIAM Journal on Imaging Sciences* (2012).

Jim Douglas and H. H. Rachford. 1956. On the Numerical Solution of Heat Conduction Problems in Two and Three Space Variables. *Trans. Amer. Math. Soc.* (1956).

H.W. Engl, M. Hanke, and A. Neubauer. 2000. *Regularization of Inverse Problems.*

Xianfeng Gu, Steven J. Gortler, and Hugues Hoppe. 2002. Geometry Images. *ACM Trans. Graph.* (2002).

P. Hansen. 2010. *Discrete Inverse Problems.* Society for Industrial and Applied Mathematics.

Felix Heide, Steven Diamond, Matthias Nieundefinedner, Jonathan Ragan-Kelley, Wolfgang Heidrich, and Gordon Wetzstein. 2016. ProxImaL: Efficient Image Optimization Using Proximal Algorithms. *ACM Trans. Graph.* (2016).

Carl W. Helstrom. 1967. Image Restoration by the Method of Least Squares. *J. Opt. Soc. Am.* (1967).

Amir Hertz, Sharon Fogel, Rana Hanocka, Raja Giryes, and Daniel Cohen-Or. 2019. Blind Visual Motif Removal from a Single Image. *CVPR* (2019).

Suwicha Imnang. 2013. Viscosity iterative method for a new general system of variational inequalities in Banach spaces. *Journal of Inequalities and Applications* (2013).

Meiguang Jin, Zhe Hu, and Paolo Favaro. 2019. Learning to Extract Flawless Slow Motion From Blurry Videos. In *CVPR.*

S. Jose, N. Mohan, V. Sowmya, and K. P. Soman. 2017. Least square based image deblurring. In *ICACCI.*

Duangkamon Kitkuan, Poom Kumam, Anantachai Padcharoen, Wiyada Kumam, and Phatiphat Thounthong. 2019. Algorithms for zeros of two accretive operators for solving convex minimization problems and its application to image restoration problems. *J. Comput. Appl. Math.* (2019).

Guillaume Lavoué and Rafal Mantiuk. 2015. Quality Assessment in Computer Graphics. *Visual Signal Quality Assessment.*

Jing Li. 2019. A Blur-SURE-Based Approach to Kernel Estimation for Motion Deblurring. *Pattern Recognition and Image Analysis* (2019).

P. Lions and B. Mercier. 1979. Splitting Algorithms for the Sum of Two Nonlinear Operators. *SIAM J. Numer. Anal.* (1979).

Lishan Liu, Genaro Lopez, Victoria Martin-Marquez, Fenghui Wang, and Hong-Kun Xu. 2012. Forward-Backward Splitting Methods for Accretive Operators in Banach Spaces. In *Abstract and Applied Analysis.*

Yiming Liu, Jue Wang, Sunghyun Cho, Adam Finkelstein, and Szymon Rusinkiewicz. 2013. A No-Reference Metric for Evaluating the Quality of Motion Deblurring. *ACM Transactions on Graphics* (2013).

Dirk A. Lorenz and Thomas Pock. 2015. An Inertial Forward-Backward Algorithm for Monotone Inclusions. *Journal of Mathematical Imaging and Vision* (2015).

Rafal Mantiuk, Kil Joong Kim, Allan G. Rempel, and Wolfgang Heidrich. 2011. HDR-VDP-2: A Calibrated Visual Metric for Visibility and Quality Predictions in All Luminance Conditions. *ACM Trans. Graph.* (2011).

R. Mantiuk, R. Mantiuk, A. Tomaszewska, and W. Heidrich. 2009. Color correction for tone mapping. *Computer Graphics Forum* (2009).

A. Mosleh, P. Green, E. Onzon, I. Begin, and J. M. P. Langlois. 2015. Camera intrinsic blur kernel estimation: A reliable framework. In *CVPR.*

A. Moudafi and M. Oliny. 2003. Convergence of a splitting inertial proximal method for monotone operators. *J. Comput. Appl. Math.* (2003).

James Nagy, Katrina Palmer, and Lisa Perrone. 2004. Iterative Methods for Image Deblurring: A Matlab Object-Oriented Approach. *Numerical Algorithms* (2004).

Seungjun Nah, Sanghyun Son, and Kyoung Mu Lee. 2019. Recurrent Neural Networks With Intra-Frame Iterations for Video Deblurring. In *CVPR.*

Neal Parikh and Stephen Boyd. 2014. Proximal Algorithms. *Found. Trends Optim.* (2014).

Withun Phuengrattana and Suthep Suantai. 2011. On the rate of convergence of Mann, Ishikawa, Noor and SP-iterations for continuous functions on an arbitrary interval. *J. Comput. Appl. Math.* (2011).

Fred Pighin and J. P. Lewis. 2007. Practical Least-Squares for Computer Graphics. In *ACM SIGGRAPH Courses.*

Boris T Polyak. 1964. Some methods of speeding up the convergence of iteration methods. *U. S. S. R. Comput. Math. and Math. Phys.* (1964).

Boris T Polyak. 1987. Introduction to Optimization. Optimization Software. *Optimization Software* (1987).

Lewis Fry Richardson. 1910. On the approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. *Proc. R. Soc. Lond. A* (1910).

R. Rockafellar. 1976. Monotone Operators and the Proximal Point Algorithm. *SIAM Journal on Control and Optimization* (1976).

Predrag Stanimirovic, Igor Stojanovic, Vasilios Katsikis, Dimitrios Pappas, and Zoran Zdravev. 2015. Application of the Least Squares Solutions in Image Deblurring. *Mathematical Problems in Engineering* (2015).

Suthep Suantai. 2005. Weak and strong convergence criteria of Noor iterations for asymptotically nonexpansive mappings. *J. Math. Anal. Appl.* (2005).

Wataru Takahashi. 2010. Fixed point theorems for new nonlinear mappings in a Hilbert space. *Journal of Nonlinear and Convex Analysis* (2010).

Matthew Trentacoste, Rafal Mantiuk, and Wolfgang Heidrich. 2011. Blur-Aware Image Downsampling. *Computer Graphics Forum* (2011).

P. Tseng. 2000. A Modified Forward-Backward Splitting Method for Maximal Monotone Mappings. *SIAM Journal on Control and Optimization* (2000).

C. Vogel. 2002. *Computational Methods for Inverse Problems.* SIAM.

Ruxin Wang and Dacheng Tao. 2014. Recent Progress in Image Deblurring. *CoRR* (2014).

Zhou Wang and Alan C. Bovik. 2002. A universal image quality index. *IEEE Signal Processing Letters* (2002).

Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. 2004. Image Quality Assessment: From Error Visibility to Structural Similarity. *Trans. Img. Proc.* (2004).

Krzysztof Wolski, Daniele Giunchi, Nanyang Ye, Piotr Didyk, Karol Myszkowski, Radosław Mantiuk, Hans-Peter Seidel, Anthony Steed, and Rafał K. Mantiuk. 2018. Dataset and Metrics for Predicting Local Visible Differences. *ACM Trans. Graph.* (2018).

Wufeng Xue, Lei Zhang, Xuanqin Mou, and Alan C. Bovik. 2014. Gradient Magnitude Similarity Deviation: A Highly Efficient Perceptual Image Quality Index. *Trans. Img. Proc.* (2014).

APPENDIX

Alvarez's Lemma. Let $\{\psi_n\}$, $\{\delta_n\}$ and $\{\alpha_n\}$ be the sequences in $[0, +\infty)$ such that $\psi_{n+1} \leq \psi_n + \alpha_n (\psi_n - \psi_{n-1}) + \delta_n$ for all $n \geq 1$, $\sum_{n=1}^{\infty} \delta_n < +\infty$ and there exists a real number α with $0 \leq \alpha_n \leq \alpha < 1$ for all $n \geq 1$. Then, the followings hold [Alvarez and Attouch 2001]:

$$(i) \quad \sum_{n \geq 1} [\psi_n - \psi_{n-1}]_+ < +\infty, \text{ where } [t]_+ = \max\{t, 0\}$$

$$(ii) \quad \exists \psi^* \in [0, +\infty) \text{ such that } \lim_{n \to +\infty} \psi_n = \psi^*.$$

Opial's condition. A given space $\mathbb{X}$ satisfies the Opial's condition if when sequence $\{x_n\}$ converges weakly to x, then

$$\liminf_{n \to \infty} \|x_n - x\| < \liminf_{i \to \infty} \|x_n - y\|$$

for all $y \in \mathbb{X}$ with $y \neq x$. It is well-known that every Euclidean space $\mathbb{R}^N$ satisfies Opial's condition.

Proximal Operator. Proximity operators are important tools serving as basic building blocks of proximal splitting algorithms. The proximal operator is used to approximate at a given value, without compromising between the accuracy of the approximation. The

proximal operator $\text{prox}_\lambda(x)$ is defined as follows:

$$\text{prox}_\lambda(x) = \begin{cases} x + \lambda & \text{if } x < -\lambda \\ 0 & \text{if } -\lambda \leq x \leq \lambda \\ x - \lambda & \text{if } x > \lambda \end{cases}$$

Given a function $g(x)$ admits sub-gradients at x_0. The set of all sub-gradients is called sub-differential at x_0 and denoted by ∂g_{x_0}. Sub-differential is always convex compact set (see Figure 14).
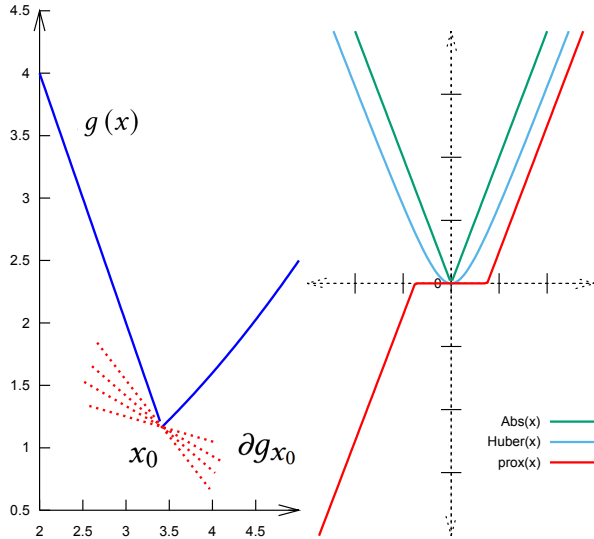

Fig. 14. Sub-gradient and Proximity operator

Gaussian Point Spread Function. In this paper, we use a collection of non-Gaussian and Gaussian point spread functions. As a simple example (as seen in Figure 15), the Gaussian PSF matrix is a two-dimensional Gaussian function, where each entry $\text{psf}(x, y)$ is defined as follows:

$$\text{psf}(x, y) = a \exp\left(-\left(\frac{(x - x_0)^2}{2\sigma_x^2} + \frac{(y - y_0)^2}{2\sigma_y^2}\right)\right)$$

where the coefficient A is the amplitude, (x_0, y_0) is the center and σ_x, σ_y are the x and y spreads of the blob. The provided 3D plot was generated using $a = 1$, $x_0 = 0$, $y_0 = 0$, $\sigma_x = \sigma_y = 1$.

Convergence of Iterative Method. The basic idea of any iterative methods is to construct a sequence of vectors $x^{(k)}$ integrating the following property of convergence proof-theoretically:

$$x = \lim_{k \to \infty} x^{(k)}$$

where x is the solution. However, the exact mathematical solution is not always reachable through the prism of a computational framework. Then, we introduce a suitable stopping criteria ensuring convergence of the iterations. To ensure that the iterative process is stopped at the minimum value of n such that $\left\| x^{(n)} - x \right\| < \epsilon$, where ϵ is a fixed tolerance and $\|\cdot\|$ is any convenient vector norm.
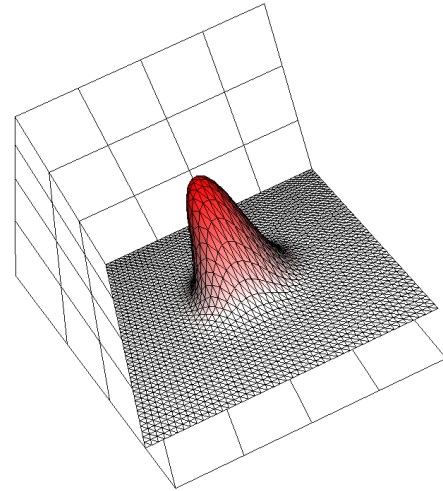
Fig. 15. Gaussian Point Spread Function

High Dynamic Range Imaging. We develop an application to filter images in the high dynamic range [Chaurasiya and Ramakrishnan 2013]. We performed deblurring in the RGBE space and then show high dynamic range tone mapping results of the deblurred HDR images for various compression and saturation tone mapping [Mantiuk et al. 2009]. Red-green-blue components are encoded using half-precision floating point number. The half-precision float offers flexibility of the floating point numbers. This is well-suited for encoding linear luminance and radiance values, as they can easily encompass large dynamic range. RGBE OpenEXR format. The RGBE pixel encoding is used in the radiance file format (32-bit per pixel RGBE encoding). The RGBE pixel encoding represents colors using four bytes. The first three bytes encode red, green, blue color channels and the last byte is a common exponent for all channels. RGBE is essentially a custom floating point representation of pixel values, which uses 8 bits to represent exponent and another 8 bits to represent mantissa. The RGBE encoding takes advantage of the strong correlation of color channels in the RGB color spaces and their values are at least of the same order of magnitude. Therefore, there is no need to store a separate exponent for each color channel.

The conversion from (R, G, B, E) bytes to red, green and blue trichromatic color values (r, g, b) is done using the formulas:

$$(r, g, b) = \begin{cases} \left(\frac{(R,G,B)+0.5}{256}\right) \cdot 2^{E-128}\left(\frac{\text{exposure}}{E_w}\right) & \text{if } E \neq 0 \\ (0, 0, 0) & \text{if } E = 0 \end{cases}$$

where exposure parameter (one for the entire image) can be used to adjust absolute values and E_w is the efficacy of the white constant equal to 179. Both these terms are used in the Radiance file format but are often emitted in other implementations. The inverse transformation is given by:

$$E = \begin{cases} \lceil \log_2(\max\{r, g, b\}) + 128 \rceil & \text{if } (r, g, b) \neq 0 \\ 0 & \text{if } (r, g, b) = 0 \end{cases}$$

$$(R, G, B) = \left\lfloor \frac{256\,(r, g, b)}{2^{E-128}} \right\rfloor$$

where $\lceil \cdot \rceil$ denotes rounding up to the nearest integer and $\lfloor \cdot \rfloor$ rounding down to the nearest integer. The limitation of the RGBE encoding is the impossibility of representing highly saturated colors. Color components become negative when highly saturated colors are converted to the RGB color space. Some color information can be lost since the RGBE format can contain negative values. To solve this problem, the radiance format can also encode pixels in the CIE XYZ color space using XYZE encoding. Such encoding is analogous to RGBE, except that CIE XYZ color primaries are used.

Skeletal Motion Data. We develop an application to deblur skeletal motion data as shown in Figure 13. We assume motion capture data have the standard skeletal format of n bones and k frames. A vector of parameters represents each frame.

$$\mathcal{M}(t) = (\boldsymbol{p}_R(t), \boldsymbol{q}_0(t), \cdots, \boldsymbol{q}_n(t), \boldsymbol{o}_0(t), \cdots, \boldsymbol{o}_n(t))$$

where $\boldsymbol{p}_R$ is a 3-vector indicating the root position in world coordinates, $\boldsymbol{q}_i$ is the quaternion specifying the orientation of the ith joint in its parent's coordinate system, and $\boldsymbol{o}_i$ is a 3-vector defining the offset of the ith joint in its parent's coordinate system. Using a linear indexing, the jth frame's vector is denoted by F_j. A motion is a continuous function $M(t)$ regularly sampled into frames $F_j = M(t_j)$, where each frame is a skeletal pose defined by its joint orientations and the position of the root joint. We cast the motion deblurring problem as a 1D signal restoration problem by considering a 1D convolution for an input signal $f[k]$ separately for each joint. Then, the $k \times k$ blurring matrix A is directly the Toeplitz matrix. For circular convolution, the Toeplitz is constant along the diagonals. In 1D, the entries $\{a_{ij}\}$ of A are related to the PSF $\mathcal{K}$ as follows:

$$a_{ij} = \mathcal{K}\left[(i - j) \mod k\right] , \ \forall i, j \in [1 \cdots k]$$