Investigating Task-driven Latent Feasibility for Nonconvex Image Modeling

Risheng Liu, Member, IEEE, Pan Mu, Jian Chen, Xin Fan, Member, IEEE, and Zhongxuan Luo

Abstract—Properly modeling the latent image distributions always plays a key role in a variety of low-level vision problems. Most existing approaches, such as Maximum A Posterior (MAP), aimed at establishing optimization models with prior regularization to address this task. However, designing sophisticated priors may lead to challenging optimization model and time-consuming iteration process. Recent studies tried to embed learnable network architectures into the MAP scheme. Unfortunately, for the MAP model with deeply trained priors, the exact behaviors and the inference process are actually hard to investigate, due to their inexact and uncontrolled nature. In this work, by investigating task-driven latent feasibility for the MAP-based model, we provide a new perspective to enforce domain knowledge and data distributions to MAP-based image modeling. Specifically, we first introduce an energy-based feasibility constraint to the given MAP model. By introducing the proximal gradient updating scheme to the objective and performing an adaptive averaging process, we obtain a completely new MAP inference process, named Proximal Average Optimization (PAO), for image modeling. Owning to the flexibility of PAO, we can also incorporate deeply trained architectures into the feasibility module. Finally, we provide a simple monotone descent-based control mechanism to guide the propagation of PAO. We prove in theory that the sequence generated by both our PAO and its learning-based extension can successfully converge to the critical point of the original MAP optimization task. We demonstrate how to apply our framework to address different vision applications. Extensive experiments verify the theoretical results and show the advantages of our method against existing state-of-the-art approaches.

Index Terms—Low-level vision, nonconvex image modeling, task-driven feasibility, maximum a posterior.

I. INTRODUCTION

While in many vision tasks we often formulate the inverse problems that finding the latent image \( x \in \mathbb{R}^D \) as \( b = Ax + n \), where \( A \) denotes some degradation matrix relating to an imaging/degradation system (such as blur kernel, downsamples or mask, etc.) and \( n \in \mathbb{R}^n \) is an i.i.d white Gaussian noise term with unknown standard noise level. Typically, in most real-world scenarios, solving the inverse problems is challenging and mathematically ill-posed, that is, the optimal solution either not exists, or is not unique. Over the past decades, numerous methods have been developed to address these low-level vision problems, such as optimizing designed-priors [1], [2], [3], [4], [5], deep learning-based approaches [6], [7], [8], [9], [10], [11], [12]. These inverse problems are often formulated as Maximum A Posterior (MAP) estimation with some conditional probability \( p(b|x) \) and prior distribution \( p(x) \), which can be \( p(x|b) \propto p(b|x)p(x) \). Indeed, solving the MAP framework is equivalent to deal with the minimization problem who tries to figure out an optimal solution subject to a certain set of constraints such that their objectives reach the best. Following this perspective, the MAP model can be formulated as

\[
\min_{x \in \mathbb{R}^D} F(x) = f(x) + \psi(x),
\]

where function \( f \) and \( \psi \) typically capture the loss of data fitting and the regularization, respectively. In this work, we assume that the loss \( f \) is smooth, while the regularization \( \psi \) can be nonconvex and nonsmooth.

Owing to the ill-posed nature of most image processing tasks, it is necessary to design priors for getting desired solutions. For example, many image restoration tasks utilize the sparsity prior as the regularization term [1], [2]. In particular,
Eq. 1 can be minimized by a broad class of general numerical optimization methods, among which the Proximal Gradient (PG) [13], Half Quadratic Splitting (HQS) [14] and Alternating Direction Method of Multipliers (ADMM) [15] are proven to be the most reliable methods. Over the past decades, many efforts have been devoted to these schemes. For example, by integrating Nesterov’s accelerated gradient method [16] into the fundamental PG scheme, APG is initially developed for convex models [1, 17]. Subsequently, other typical APGs are derived solving problem 1 including monotone APG (mAPG) [18], inexact APG (niAPG) [19] and momentum APG for nonconvex problem (APGnc) [20], etc. Optimization designed priors strategies provide a mathematical understanding of their behaviors with well-defined regularization properties. Flexible and exact proper prior is challenging to construct and solve. However, simple regularizer performs poorly when compared with state-of-the-art methodologies in real-world applications. This is because these methods do not exploit the particular structures of the image processing tasks at hand nor the input data distributions. These limits make it difficult to solve the problems in a purely optimized manner with designed priors.

Different from the optimizing designed-priors schemes, learning-based approaches learn mapping functions to deduce the desirable high-quality images from the observed one. In recent years, various learning-based strategies [6], [9], [10] have been proposed to address practical image modeling problems. These discriminative learning-based methods combine the classical numerical solvers and the collected training data to obtain some task-specific iterations. Similar to this view, plug-in schemes have been recently studied extensively with a great empirical success that replace the regularization term by using task-related operator for vision problems [21], [22], [23], [24]. Indeed, these algorithms perform better than some state-of-the-art methods in real-world applications. Unfortunately, these plug-in schemes consider the prior term with implicit regularization which may break the properties and structures of the objective in Eq. 1. Thus, the existing proofs only demonstrate the iteration sequences converge to a fixed point without knowing the relationship between it and the optimal solutions. By introducing spectral normalization technique for more accurately constraining deep learning-based denoisers, the fixed-point theory is established in [25]. Whereas this theory result is effective only under the strongly convex condition of function \( f(x) \) which is unsatisfied in plenty of vision problems, for example when the data fitting term \( f(x) \) is setting as \( \frac{1}{2} \| Ax - b \|_2^2 \) with non-column full rank matrix \( A \), the strong convex property is unattainable. In contrast to these implicit plug-and-play methods, an explicit plug-in scheme is developed in [17], named Regularization by Denoising (RED), with explicit Laplacian-based regularization functional. While the regularization term is required to be symmetric which is unsatisfied in many state-of-the-art methods, such as NLM [26], RF [27] and BM3D [28] etc. Further work has been discussed in [29], [30]. Optimal condition-based approaches, such as [31], [32] have been developed efficiently for solving Eq. 1. However, the implicit condition relies on estimating the sub-differential which is usually attained implicitly.

To address the above issues, we develop a Proximal Averaged Optimization (PAO) for the challenging MAP-based nonconvex and nonsmooth problem described in Eq. 1. The proposed PAO is a joint process of optimizing objective and feasibility constraint. Specifically, by enforcing task-driven latent feasibility for MAP-type model, we develop a new perspective to investigate domain knowledge and data distributions for image modeling (see Fig. 1 for illustration.). For Task-driven Feasibility (TF) type minimization problem, we establish a TF-PAO scheme. Considering the flexibility of PAO, we further embed the learnable form into the feasibility constraint and derive a LTF-PAO scheme. The learnable form incorporates designed/trained architectures that aim to find the task-related optimal solution. We rigorously prove that both the proposed PAO and its learning-based extension scheme converge to a critical point of the original problem 1 with a commonly used monotone descent condition. We also demonstrate how to apply our paradigm to address challenging real-world vision applications, such as image deblurring, inpainting and rain streaks removing. Different from manually designed-priors solved by general numerical methods, the proposed PAO exploits the particular structures and the data distributions of image processing. Learning-based methodologies [21], [23], [24], [8] embed learnable network architectures into the MAP framework, whereas the inference process and the exact behaviors of which are actually hard to investigate. In comparison, the developed PAO enables the MAP framework to insert the learnable structures without changing the properties of the objective. In summary, the contributions of this paper mainly include:

- The developed PAO provides a novel perspective to introduce task-driven feasibility module for nonconvex and nonsmooth MAP-based image modeling problems. By embedding the additional learning-based deep architectures, PAO exploits the data distribution and the particular structure by incorporating the LTF module.
- The developed TF-PAO and LTF-PAO keep the properties and structures of the objective. Specifically, the developed iteration sequences converge to one of the critical points of Eq. 1. Moreover, we prove in theory that the proposed frameworks can derive sequence convergence results.
- Further, we also consider the PAO as a flexible ensemble framework to solve the optimization model as described in Eq. 1 when addressing different real-world computer vision tasks. Extensive experiments show the superiority of our PAO method on the tested problems.

II. THE PROPOSED ALGORITHM

In this section, by enforcing task-driven feasibility, we first reformulate the minimization problem 1 to a constraint-based scheme which can embed TF and LTF modules for vision tasks. Then, by incorporating with LF and LTF, the PAO is developed for the nonconvex modeling-based image restoration problem. The convergence behaviors about the PAO are performed under some loose conditions about the objective function \( F(x) \) accordingly.
A. Enforcing Task-driven Feasibility

To solve the MAP-based nonconvex minimization problem as described in Eq. (1), we first reformulate it by enforcing task-driven feasibility module. Then, the original problem can be reformulated as the following constraint-based form

$$
\min_{x} F(x), \text{ s.t. } x \in \mathcal{G}.
$$

(2)

In general, $\mathcal{G}$ is designed as a rough estimation, such as, the bounded domain $[0, 1]^n$, or a set estimated by some equality or inequality constraints, which is normal to characterize the solution space in image processing problems \cite{13, 34}. In this work, the energy-based latent feasibility constraint can be constructed by the TF module, named $\mathcal{G}_{TF}$, for image restoration problems,

$$
\mathcal{G}_{TF}(x) \in \arg \min_{x} \{g(x) + \phi(x)\},
$$

(3)

where $g(x)$ is differentiable and $\phi(x)$ is nonsmooth and nonconvex. We assume that $g(x)$ and $\phi(x)$ are proper and lower-semicontinuous.

Considering the important feature of the complex data distribution in real-world applications, we further introduce Learning-based TF (LTF) module to incorporate designed/trained architectures to optimize Eq. (1). Specifically, the network-based building block at the $k$-th iteration can be denoted as $\mathcal{N}(\cdot; \mathcal{W}_T^k)$, where $\mathcal{W}_T = \{\mathcal{W}_T^k\}_{k=0}^T$ is the set of learnable parameters with $T$-th training stage.\cite{14} We denote the temporary variable at the $k$-th iteration as $\tilde{x}^k = \mathcal{N}(x^k, \mathcal{W}_T^k)$. By further considering LTF form as the proximal approximation of Eq. (3) with parameter $\mu_k > 0$ at the $k$-th iterative, the learning-based-task-driven feasibility module $\mathcal{G}_{LTF}$ can be

$$
\mathcal{G}_{LTF}^\mu(x, \tilde{x}^k) \in \arg \min_{x} \{g(x) + \phi(x) + \mu_k/2\|x - \tilde{x}^k\|^2\}.
$$

(4)

Indeed, the modules $\mathcal{G}_{TF}$ and $\mathcal{G}_{LTF}$ are task-related which either can be selected to maintain some of the characteristics of the problem, such as, smooth, edge-enhanced, denoiser and sparse, etc., or can be introduced to characterize the task on another domain. For example, the argument of minimized sparse, etc., or can be introduced to characterize the task of the problem, such as, smooth, edge-enhanced, denoiser and either can be selected to maintain some of the characteristics nonconvex. We assume that $\mathcal{G}_{TF}(x)$ is nonsmooth and

$$
\mathcal{G}_{TF}(x) = \arg \min_{x} \{g(x) + \phi(x) + \mu_k/2\|x - \tilde{x}^k\|^2\}.
$$

(4)

where $g(x)$ is differentiable and $\phi(x)$ is nonsmooth and nonconvex. We assume that $g(x)$ and $\phi(x)$ are proper and lower-semicontinuous.

Considering the important feature of the complex data distribution in real-world applications, we further introduce Learning-based TF (LTF) module to incorporate designed/trained architectures to optimize Eq. (1). Specifically, the network-based building block at the $k$-th iteration can be denoted as $\mathcal{N}(\cdot; \mathcal{W}_T^k)$, where $\mathcal{W}_T = \{\mathcal{W}_T^k\}_{k=0}^T$ is the set of learnable parameters with $T$-th training stage.\cite{14} We denote the temporary variable at the $k$-th iteration as $\tilde{x}^k = \mathcal{N}(x^k, \mathcal{W}_T^k)$. By further considering LTF form as the proximal approximation of Eq. (3) with parameter $\mu_k > 0$ at the $k$-th iterative, the learning-based-task-driven feasibility module $\mathcal{G}_{LTF}^\mu$ can be

$$
\mathcal{G}_{LTF}^\mu(x, \tilde{x}^k) \in \arg \min_{x} \{g(x) + \phi(x) + \mu_k/2\|x - \tilde{x}^k\|^2\}.
$$

(4)

Indeed, the modules $\mathcal{G}_{TF}$ and $\mathcal{G}_{LTF}$ are task-related which either can be selected to maintain some of the characteristics of the problem, such as, smooth, edge-enhanced, denoiser and sparse, etc., or can be introduced to characterize the task on another domain. For example, the argument of minimized objective function $\arg \min_x F(x)$ and the constraint module $\mathcal{G}_{TF}$ can be constructed as the image domain and the gradient domain respectively.

B. Proximal Averaged Optimization

In the light of the model (2), two task-driven feasibility type (TF and LTF) PAO are developed in this section.

1) PAO with Task-driven Feasibility: Specifically, for objective minimization problem (1), we just adopt PG method to update the variable which yields

$$
\mathcal{F}_t(x) = \text{prox}_{\tau t}(x - \tau \nabla f(x)),
$$

(5)

where $\text{prox}_{\tau t}(\cdot) = \arg \min_{x} \{\psi(x) + \frac{1}{2}\|x - \cdot\|^2\}$ and $t$ is the step size. As for the Eq. (5), we actually perform any first-order methods to solve this subproblem such as, PG \cite{11},

Algorithm 1 Task-driven Feasibility PAO (TF-PAO)

**Require:** The input $x^0$, $\alpha^0$, parameters $t_k \in (0, 1/L_t)$ and $\gamma \in (0, 1).

1: while not converged do
2: \quad $x^k_T = \mathcal{F}_{t_k}(x^k)$ and $x^k_G = \mathcal{G}_{TF}(x^k)$.
3: \quad $v^k = \alpha^k x^k_T + (1 - \alpha^k)x^k_{G}$.
4: \quad $x^{k+1} = \text{MDUS}(v^k, x^k_T; \alpha^k, \gamma)$.
5: end while

Algorithm 2 $x^{k+1} = \text{MDUS}(v^k, x^k_T; \alpha^k, \gamma)$

1: if $F(v^k) \leq F(x^k_T)$ then
2: \quad $x^{k+1} = v^k$.
3: else
4: \quad $x^{k+1} = x^k_T$.
5: end if
6: $\alpha^{k+1} = \gamma \alpha^k$.

2) PAO with Learning-based Task-driven Feasibility: By embedding designed/trained architectures to TF module, we then develop the LTF-PAO to optimize MAP-based model described in Eq. (1). What share with the statement in subsection II-B is the freedom for selecting method to solve this constraint-based subproblem. Then by introducing a relative loose boundness condition about $\mathcal{G}_{LTF}^\mu(x, \tilde{x}^k)$, we summarize the complete LTF-PAO scheme in Alg. 3. Indeed, $\tilde{x}^k$ is the output of network $\mathcal{N}$ which is used to approximate the task-driven module $\mathcal{G}_{TF}(x)$ at the $k$-th iteration. We introduce a boundness condition (in Steps 3 of Alg. 3) to control the iteration sequence. This aims to prevent any improperly designed/trained architectures, which may deflect our iterative trajectory towards unwanted solutions. The monitor is obtained by checking the boundedness of $\|x^k - x^k_T\|$. The convergence behaviors are summarized in the following section.

III. CONVERGENCE RESULTS

In this part, we would like to discuss the convergence behaviors for the proposed TF-PAO and LTF-PAO algorithms.
We suggest readers to refer to [37] for some definitions in variational analysis, such as proper, lower-semicontinuous, coercive and the limiting subdifferential which will be useful in the following analysis. Our convergence analyses are also based on the following fairly loose assumptions.

**Assumption 1.** The objective function $F(x)$ in Eq. (1) is proper, lower-semicontinuous and coercive. Function $f(x)$ is convex and Lipschitz smooth, i.e., $\forall x, y \in \mathbb{R}^p$, we have $\|\nabla f(x) - \nabla f(y)\| \leq L_f \|x - y\|$, where $L_f$ is the Lipschitz constant for $\nabla f$.

**Theorem 1.** Let $\{x^k, x^*_k, x^{k+1}\}_{k \in \mathbb{N}}$ be the iteration sequences generated by Alg. 1. Then the results are summarized in the following.

- There exists a constant $\sigma > 0$ satisfying that $F(x^{k+1}) \leq F(x^k) - \sigma \|x^k - x^k\|^2$.
- Let $x^k$ be any accumulation of the sequence $\{x^{k+1}\}$ which implies that $x^k \!$ is a critical point of the minimization problem in Eq. (1), i.e., $0 \in \partial F(x^k)$.

**Proof.** We first show the sufficient descent property about $F(x)$. By using the proximal update scheme in Step 2 of Alg. 1 and the Lipschitz property of $f(x)$, i.e.,

$$\psi(x^k) + \langle x^k - x^* \rangle, \nabla f(x^k) \rangle + \frac{1}{2\beta} \|x^k - x^k\|^2 \leq \psi(x^k) \text{ and } f(x^k) \leq f(x^k) + \langle x^k - x^k \rangle, \nabla f(x^k) \rangle + \frac{L_f}{2}\|x^k - x^k\|^2,$$

we conclude that $F(x^k) \leq F(x^k) - \left(\frac{1}{\beta} - \frac{L_f}{2}\right)\|x^k - x^k\|^2$. From the MDUS correction scheme, the sufficient descent property is obtained with $\sigma = \frac{1}{\beta} - \frac{L_f}{2}$. Then, we will show the second item. The sufficient descent inequality in the first item implies $\sum_{k=1}^{\infty} \|x^k - x^k\|^2 < \infty$, which means that $\|x^k - x^k\| \rightarrow 0, k \rightarrow \infty$, i.e., there exists subsequence $\{x^k\}_k$ and $\{x^k\}_k$ convergence to a same point $x^*$ as $p \rightarrow \infty$. Incorporating the lower-semicontinuous of $F(x)$ and the supreme principle, we obtain that $\lim_{p \rightarrow \infty} F(x^k) = F(x^*)$. With the optimal condition $0 \in \partial \psi(x^k) + \nabla f(x^k) + \frac{1}{2\beta} \langle x^k - x^k \rangle$, we know that $-\frac{1}{\beta} \langle x^k - x^k \rangle - \nabla f(x^k) + \nabla f(x^k) \in \partial F(x^k)$. Actually, for $k \rightarrow \infty$, we have $\frac{1}{\beta} \langle x^k - x^k \rangle + \nabla f(x^k) - \nabla f(x^k) \leq \left(\frac{1}{\beta} + L_f\right)\|x^k - x^k\|^2 \rightarrow 0$. The above imply that $x^k \!$ is a critical point. This complete the proof.

Note that, the proposed algorithm is a modification PG scheme. Under some mild conditions, for example, the semi-algebraic property on $F(x)$, the sequence convergent property $\{x^k\}$ still holds.

**Theorem 2.** With the semi-algebraic property$^3$ of function $F(x)$, we can further assert that the sequence $\{x^k\}_{k \in \mathbb{N}}$ in Alg. 1 has finite length, i.e., $\sum_{k=1}^{\infty} \|x^k - x^*\| < \infty$.

**Proof.** With the KL property (see [38]) and the definition of the sub-differential, we have

$$\varphi'(F(x^k) - F(x^*)) \geq \varphi'(F(x^k) - F(x^*)) (F(x^k) - F(x^*)) \geq \frac{1}{1+\varepsilon L_f} F(x^k) - F(x^k),$$

If we denote $\Delta_{k+1} = \varphi(F(x^k) - F(x^*)) - \varphi(F(x^k) - F(x^*)) + \varepsilon L_f (F(x^k) - F(x^*))$, we obtain $\sum \|x^k - x^k\| < \infty$. Otherwise, $\sum \|x^k - x^k\| < \infty$ holds which implies $2\|x^k - x^k\| \leq \|x^k - x^k\| + \frac{\varphi(F(x^k) - F(x^*))}{1+\varepsilon L_f}$. Subsequently, we have

$$2 \sum_{i=1}^{k} \|x^k - x^k\| \leq \sum_{i=1}^{k} \|x^k - x^k\| + \frac{\varphi(F(x^k) - F(x^*))}{1+\varepsilon L_f} \sum _{i=1}^{k} \|x^k - x^k\| < \infty.$$

Obviously, the above inequality implies the finite length of sequence $\{x^k - x^k\}$. If $x^k = x^k$, the inequality $\sum \|x^k - x^k\| < \infty$ holds. If $x^k = x^k + (1 - \alpha) x^k$, with the proper, lower-semicontinuous and coercive property of $F$, the sequence $\{x^k\}$ is bounded. Then with the update scheme about $\alpha$ and the finite length of $\{x^k - x^k\}$, we have

$$\sum_{k=1}^{\infty} \|x^k - x^k\| \leq \sum_{k=1}^{\infty} \left(\alpha^k \|x^k - x^k\| + (1 - \alpha) \|x^k - x^k\|\right) < \infty.$$

This completes the proof.

Indeed, the summable sequence $\{\|x^k - x^k\|\}_{k \in \mathbb{N}}$ as stated in Theorem 2 implies that there exist $m > n > l$ satisfying $\|x^m - x^m\| < \sum_{k=n}^{m} \|x^k - x^m\| \rightarrow 0$, as $l \rightarrow \infty$. Subsequently, it follows that the iteration $\{x^k\}_{k \in \mathbb{N}}$ is a Cauchy sequence and hence is a globally convergent sequence which is also defined as sequence convergence.

**Remark 1.** According to the convergence analysis described in [37], the objective function $F(x)$ is sufficiently descent in Alg. 1 and it is easy to check that the convergence results of Alg. 1 can be obtained in the same manner as stated in Theorem 2. The temporary iteration $x^k$ in Alg. 1 is bounded under the checking condition. This implies the boundedness of $\|x^k - x^k\|$. Then, in Alg. 1, the sequence convergent property of $\{x^k\}$ is attained.

---

$^3$Please refer to [38] for the formal definition of semi-algebraic function. Actually, many functions arising in learning and vision areas, including $f_0$ norm, rational $\ell_p$ norms (i.e., $p = p_1/p_2$ with positive integers $p_1$ and $p_2$) and their finite sums or products, are all semi-algebraic.
IV. APPLICATIONS

We emphasize that different from these existing image modeling approaches, the proposed PAO allows us to introduce a task-driven feasibility module related to the application areas when solving the optimization model in Eq. (1). This section first considers non-blind deconvolution and image inpainting. We take non-blind deconvolution as an illustrative example for establishing PAO. Then, we extend the PAO to even more challenging single image rain streaks removal task.

A. Image Deconvolution

Here we consider a particular non-blind deconvolution problem, which aims to recover latent image $x$ from blurred observation $b$. By formulating this problem using sparse coding model $b = D\beta + n$, where $\beta$ denotes the sparse code, $D$ is a given dictionary, and $n$ is the unknown noise, we derive a specific case of Eq. (1), that is $\min_{\beta} \frac{1}{2}\|b - D\beta\|^2 + \lambda_1\|\beta\|_p$, where $p \in [0, 1]$, $\lambda_1 > 0$. Subsequently, it can be equivalently described as the following intuitive form, i.e.,

$$\min_{x} \frac{1}{2}\|Kx - b\|^2 + \lambda_1\|Wx\|_p.$$  

(6)

In the following, we will give an example to illustrate PAO with two module settings, i.e., task-driven feasibility $G_{TF}$ and learning-based task-driven feasibility $G_{LTF}$.

1) With Task-driven Feasibility: As for the module $G_{TF}$, we aim to introduce a relatively simple and task-related model to enforce our distribution assumptions on the latent image. Then, we introduce the widely used Total Variation (TV) model in image domain i.e.,

$$G_{TF}(x) = \arg \min_{x} \frac{1}{2}\|Kx - b\|^2 + \lambda_2 \sum_{j \in \{h, v\}} \|\nabla_j x\|_q,$$  

(7)

where $\lambda_2$ is threshold parameter, $q \in (0, 1)$, $\nabla_h$ and $\nabla_v$ respectively denote the gradient on the horizontal and vertical directions. As it is flexible to select operators for solving Eq. (7), we indeed apply HQS scheme to update $x^k_T$ in this paper. By introducing two auxiliary variables (named $z_k^h$ and $z_k^v$), the iteration $x^k_T$ is

$$x^k_T = \arg \min_{x} \frac{1}{2}\|Kx - b\|^2 + \sum_{j \in \{h, v\}} \rho_j \|z_j^k - \nabla_j x\|^2,$$

where $\rho_h > 0$ and $\rho_v > 0$ are two constant parameters. $z^h_k$ and $z^v_k$ are updated by proximal gradient operator that we omit it here. Then, applying the proximal gradient approach to update $x^k_T$ which can be transformed as $x^k_T = W^T \beta^k_T$ yields the following form

$$\beta^k_T = \text{prox}_{s_k\|\beta\|_p}(\beta^k - s_k(WK^T KW^T \beta^k - WK^T b)),$$

where $\beta^k = Wx^k$. It is clearly to obtain the detailed updating steps following Alg. 1.

2) With Learning-based Task-driven Feasibility: Specifically, in this work, the network $N(\cdot)$ is considered as a residual formulation. The learning-based iteration step can be directly adopted by $N(x^k; \mathcal{W}_{T}^k) = x^k - \sum_{t=0}^{T-1} \mathcal{U}(x^{t}; \mathcal{W}_{T}^k)$ where $\mathcal{W}_{T}^k = \{\mathcal{W}_{t}^k\}_{t=0}^T$ is the set of learnable parameters with T-th training stage and $\mathcal{U}$ is the basic network unit. In our network, there are nineteen layers which includes seven convolution layers, six ReLU layers, five batch normalization layers and one loss layer. The detailed information about $N$ can be found in the experimental results section. Notice that standard training strategies can be directly adopted to optimize parameters of our basic architecture. If necessary, discarding this section, user may further jointly tune fine-tune parameters of the whole network after the design phase. By setting $\hat{x}_T = N(x^k; \mathcal{W}_{T}^k)$, the learning-based scheme of $G^\mu_{LTF}(x, \hat{x}_T)$ is

$$G^\mu_{LTF}(x, \hat{x}_T) \in \arg \min_{x} \frac{1}{2}\|Kx - b\|^2 + \lambda_2 \sum_{j \in \{h, v\}} \|\nabla_j x\|_q + \frac{\mu}{2}\|x - \hat{x}_T\|^2.$$

Hence, following the iteration form of $x^k_T$ and the above learning-based scheme, we obtain that

$$x^k_{\mu, T} = \arg \min_{x} \frac{1}{2}\|Kx - b\|^2 + \sum_{j \in \{h, v\}} \rho_j \|z_j^k - \nabla_j x\|^2 + \frac{\mu}{2}\|x - \hat{x}_T\|^2.$$

By considering the latent image $x$ as the uniform augment, we actually obtain a PAO to integrate both synthesis and analysis mechanisms to address different vision applications, including deblurring, inpainting and rain streaks removal etc. Here the matrix $K$ actually formulates the observation forward model for particular image processing paradigm. Possible choices of $K$ include an identity operator for denoising, convolution operators for deblurring, filtered subsampling operators for superresolution, the Fourier $k$-domain subsampling operator for magnetic resonance imaging (MRI) reconstruction or mask for image inpainting. We incorporate experimentally designed and trained network architectures into the PAO for
solving these problems. In summary, the proposed PAO indeed could integrates advantages from different domain knowledge.

B. Rain Streaks Removal

This subsection focuses on single image rain streaks removal task which is a challenging real-world computer vision problem. A rainy image $y$ is often considered as linear combination of rain-free background $x_b$ and rain streaks layer $x_r$, i.e., $y = x_b + x_r$. We set $x := [x_b; x_r]$. In terms of designing the optimization model, rather than make efforts in exploiting complex priors, we consider the fundamental energy-based sparsity of the observation in certain transform domain as

$$\min_{x_b, x_r, \beta, \gamma} f(x_b, x_r; \beta, \gamma) + \psi_\beta(\beta) + \psi_\gamma(\gamma) + \chi_{[0,1]}(x_b, x_r),$$

where $f(x_b, x_r; \beta, \gamma) = \frac{1}{2} \|x_b - D \beta\|^2 + \frac{1}{2} \|x_r - D \gamma\|^2$, $\psi_\beta(\beta) = r_1 \|\beta\|_p, \psi_\gamma(\gamma) = r_2 \|\gamma\|_p$, $p_1, p_2 \in [1, \infty)$ and $r_1, r_2$ are two positive constants. $\chi_{[0,1]}$ is the indicator function, i.e., if $x_b, x_r \in [0, 1]$ then $\chi_{[0,1]}(x_b, x_r) = 0$, otherwise $\chi_{[0,1]}(x_b, x_r) = \infty$. $\beta$ and $\gamma$ respectively denote the sparse codes of $x_b$, $x_r$ on $D$ which are two auxiliary variables serving for the subproblem. As for $G_{TF}$ stated in Eq. (2), we consider the general TV regularization as described in the following form

$$G_{TF} \in \arg \min_{x_b, x_r} \frac{1}{2} \|y - x_b - x_r\|^2 + p_1 \sum_{j \in \{b, r\}} \|\nabla x_b\|_q + p_2 \|x_r\|_q.$$

In this part, we first introduce two residual network $N_b$ and $N_r$ as stated in [8]. Then we denote two temporary variable $\tilde{x}_b^k = N_b(x_b^k; \mathbf{W}_{T,b}^k)$ and $\tilde{x}_r^k = N_r(x_r^k; \mathbf{W}_{T,r}^k)$ respectively for background and rain streaks layers. For the background layer network $N_b$, we just follow [8] to build a series of denoising CNNs which extract the natural image well. For rain streaks layer, the $N_r$ learns rain streaks behavior from rainy images by training rainy image and synthetic rain layer as the degraded clean image pair. We update variables $x_{\tilde{F},b}$ and $x_{\tilde{F},r}$ synchronously

$$\begin{bmatrix} x_{\tilde{F},b}^k \\ x_{\tilde{F},r}^k \end{bmatrix} = \arg \min_{x_b, x_r} \frac{1}{2} \|x_b - D \beta^k\|^2 + \chi_{[0,1]}(x_b^k, x_r^k),$$

$$\begin{bmatrix} x_{\tilde{F},b}^k \\ x_{\tilde{F},r}^k \end{bmatrix} = \arg \min_{x_b, x_r} \frac{1}{2} \|x_r - D \gamma^k\|^2 + \chi_{[0,1]}(x_b^k, x_r^k),$$

where the auxiliary variables $\beta^k$ and $\gamma^k$ are updated by proximal gradient operator. Similarly, we obtain that

$$\begin{bmatrix} x_{\tilde{F},b}^k \\ x_{\tilde{F},r}^k \end{bmatrix} = \arg \min_{x_b, x_r} G(x_b^k, x_r^k) + \frac{1}{2} \|x_b - \tilde{x}_b^k\|^2.$$

Then, following the steps described in LTF-PAO, we obtain the updating scheme.

V. EXPERIMENTAL RESULTS

In this section, we first verify our theoretical results by investigating the iteration behaviors of the proposed TF-PAO and LTF-PAO on standard deconvolution formulation with Eq. (6). We then evaluated the state-of-the-art performance of
LTF-PAO both with general and learning-based methods on different vision applications. We conducted these experiments on a computer with Intel Core i7-7700 CPU (3.6 GHz), 32GB RAM and an NVIDIA GeForce GTX 1060 6GB GPU.

### A. Theoretical Verifications

To verify our theoretical investigations, we performed experiments on non-blind deconvolution. Notice that this problem can be directly addressed by our TF-PAO and LTF-PAO.

1) Modularization Settings: We first provide an comparison about the optimization models in image deblurring application and the corresponding results are shown in Fig. 2. Observed that the proposed PAO with LTF module performs better both than the modeling scheme in Eq. (1) and the LTF in Eq. (4). This illustrates the effectiveness of the proposed PAO scheme.

We then analyze the performance and flexibility of PAO with different operator settings and the corresponding PSNR results with 1% noise level are plotted in Fig. 3. As for solving the subproblem (7) specified in $G_{TF}$, four different first-order methods, such as PG ($G_{PG}$), APG ($G_{APG}$), HQS ($G_{HQS}$) and ADMM ($G_{ADMM}$) are considered as mentioned in subsection II-B. The PSNR scores of TF-PAO under four different strategies are plotted in Fig. 3 (a). Observed that various methods when obtaining $x^{k}_{G}$ have a slight influence on the performance of our TF-PAO scheme. We adopt HQS as the approach obtaining the iteration steps of $x^{k}_{G}$ in TF-PAO and $x^{k}_{G}$ in LTF-PAO. Note to say, to provide a relative fairness comparison, we keep parameters same under four different circumstances mentioned above. Hereafter, we select relative error (i.e., $\|x^{k+1} - x^{k}\|/\|x^{k+1}\|$) as stop criterion.

To analyze the effects of network-based block $N$, four different task-specific structures, i.e., TV [39], RF [27], CNNs and BM3D [28] (named as $N_{TV}$, $N_{RF}$, $N_{CNN}$ and $N_{BM3D}$, respectively) are adopted under the LTF-PAO scheme. For CNNs architecture, we introduce a residual network which consists of nineteen layers: seven dilated convolutions with $3 \times 3$ filter size, six ReLu operations (plugged between each two convolution layers) and five batch normalizations (plugged between convolution and ReLU, expect the first convolution layer). In training stage, we randomly select 800 natural images from ImageNet database [41]. The selected pictures are cropped into patches of size $35 \times 35$. Fig. 3 (b) plotted the PSNR with $N_{TV}$, $N_{RF}$, $N_{CNN}$ and $N_{BM3D}$. As can be seen, LTF-PAO performs better and faster with $N_{CNN}$ than others. Hence, we setting $N$ as CNNs hereafter.

We further compared our LTF-PAO with the traditional methods under three different Gaussian noise levels (i.e., 1%, 2% and 3%) on the image set (collected by [40]) and the corresponding results are shown in Tab. 1 with quantitative performance (i.e., PSNR and SSIM metrics). It can be seen that our LTF-PAO outperforms classical numerical solvers by a large margin in terms of the performance.
iteration steps. The corresponding visual results are shown that, LTF-PAO perform the best both in PSNR scores and the faster than other PGs under the same stop condition. Observed that the proposed LTF-PAO remove more noise while keeping the details.

### B. State-of-the-art Comparisons

We then evaluated our LTF-PAO on a variety of low-level vision applications including image deblurring, image inpainting and rain streaks removal.

1) **Image Deblurring:** In this task, matrix $K$ stated in the application part is the blur kernel and $b$ is blurry image. As usual, the blurry images are synthesized by applying a blur kernel and adding additive Gaussian noise. We consider the circular boundary conditions when performing the convolution. We reported the results of our LTF-PAO on Sun et al’ challenging benchmark \[43\] and Levin et al’ dataset \[44\], together with other state-of-the-art methods including the traditional methods (e.g., IDDBM3D \[45\], TV \[46\], parameters learning based methods (e.g., EPLL \[47\], TV \[46\], ) and network based methods (e.g., MLP \[48\], IRCNN \[8\], FDN \[49\]). It can be seen in Tab. \[II\] that our method obtained the best quantitative performance (i.e., PSNR and SSIM metrics) on Sun et al’ and Levin et al’ dataset. Moreover, we illustrated the visual comparisons on real image deblurring \[50\] with unknown blur kernel which is estimated roughly by Pan et al.’ method \[51\]. As shown in Fig. \[7\] our method reserve more details.

2) **Image Inpainting:** In image inpainting task, the matrix $K$ is mask, and $b$ is the missing pixels image. This task aims to recover the missing pixels of the observation. Here we compared our LTF-PAO with TV \[39\], FOE \[52\], VNL \[53\], ISDSB \[54\], WNNM \[55\] and IRCNN \[8\] on this problem. We normalized the pixel values to $[0, 1]$. We generated random masks of different levels including 40%, 60%, 80% missing pixels on CBS68 dataset \[42\]. Moreover, we illustrated the visual comparisons in Fig. \[8\] with top five scores (TV, FOE, ISDSB, WNNM and IRCNN). It can be observed that our approach successfully recovered the image with better visual quality, especially in the zoomed-in regions with rich details.

### Table III

**Averaged Quantitative Comparison of Image Inpainting on CBSD68 Dataset \[42\].** The first row is the proportion of masks. The first column is the comparison methods on inpainting.

| Mask     | 40%     | 60%     | 80%     | Text     |
|----------|---------|---------|---------|----------|
| TV       | 32.22/0.93 | 29.20/0.86 | 26.07/0.74 | 35.29/0.97 |
| FoE      | 34.01/0.90 | 30.81/0.81 | 27.64/0.65 | 37.05/0.95 |
| VNL      | 27.55/0.91 | 26.13/0.85 | 24.23/0.75 | 28.58/0.95 |
| ISDSB    | 31.32/0.91 | 28.23/0.83 | 24.92/0.70 | 34.91/0.96 |
| WNNM     | 31.75/0.94 | 28.71/0.89 | 25.63/0.78 | 34.89/0.97 |
| IRCNN    | 34.92/0.95 | 31.45/0.91 | 26.44/0.79 | 37.26/0.97 |
| Ours     | 34.94/0.96 | 31.61/0.91 | 27.88/0.81 | 37.38/0.98 |

---

Fig. 8. Image inpainting results (with PSNR / SSIM scores) on a challenging example image with 80% missing pixels.
3) Single-image Rain Streaks Removal: In this part, we evaluated our method on the task of rain streaks removal, in comparison with the state-of-the-art including GMM [34], DN [58], DDN [59], JCAS [60], JORDER [61], UGSM [62], and DID-MDN [63]. All the comparisons shown in this paper are conducted under the same hardware configuration.

Tab. IV reported the quantitative scores on three different datasets: (1) Test1 is obtained by [34], includes 12 synthesized rain images with only one type of rain streaks rendering technique. (2) Rain100H is collected from BSD200 [57] and synthesized with five streak directions; (3) Test2 data-set consists of 7 images, using photorealistic rendering of rain streaks [56]. Here we just adopt the training set provided by Yang et al. [61] for N. According to the quantitative results reported in Tab. IV, we provide visual comparisons for five methods with relative high PSNR and SSIM scores (i.e., GMM, DDN, UGSM, JORDER, DID-MDN) in Fig. 9.

![Fig. 9. Rain streaks removal comparison results of synthesized image (top row) from Test1 and real-world rain image (middle and bottom row).](image)

### TABLE IV

| Methods | Test1 | Test2 | Rain100H |
|---------|-------|-------|----------|
|         | PSNR  | SSIM  | PSNR     | SSIM    | PSNR     | SSIM    |
| JCAS    | 31.61 | 0.9183| 28.37    | 0.9050  | 15.23    | 0.5150  |
| GMM     | 32.33 | 0.9042| 29.57    | 0.8878  | 14.26    | 0.4225  |
| DN      | 30.30 | 0.9151| 27.34    | 0.9009  | 13.72    | 0.4417  |
| DDN     | 33.41 | 0.9442| 29.91    | 0.9433  | 17.93    | 0.5655  |
| UGSM    | 33.30 | 0.9253| 27.07    | 0.9220  | 14.90    | 0.4674  |
| JORDER  | 35.93 | 0.9530| 35.11    | 0.9732  | 23.45    | 0.7490  |
| DID-MDN | 29.08 | 0.9015| 27.92    | 0.8695  | 17.28    | 0.6035  |
| Ours    | 36.39 | 0.9630| 34.88    | 0.9737  | 24.30    | 0.8044  |

It can be observed that the proposed LTF-PAO scheme can reserves more details with very few rain streaks left no matter in synthesized or real-world rainy images.

### TABLE V

| Methods | PSNR | SSIM |
|---------|------|------|
| DN      | 25.51| 0.8885|
| UGSM    | 26.38| 0.8261|
| DDN     | 29.90| 0.8999|
| JORDER  | 31.18| 0.9152|
| Ours    | 34.81| 0.9531|

![Fig. 10. Image rain streaks removal results (PSNR / SSIM scores) on Fu et al.’ test set.](image)

Furthermore, we conducted experiments on a large scale dataset with 1,400 test images (collected by [59]). The quantitative and qualitative results are demonstrated in Tab. V and Fig. 10 respectively. Observed that our method performed much better than the compared ones.

### VI. Conclusions

In this paper, we developed a Proximal Averaged Optimization (PAO) method for the challenging nonconvex MAP-based
model in Eq. (1). By introducing two constraint schemes, i.e., task-driven feasibility and learning-based task-driven feasibility module, TF-PAO and LTF-PAO were established respectively. Then we proved the convergence of PAO with some relatively loose assumptions. Extensive experiments on some challenging tasks showed that our method has better visual performance and quantitative scores against other state-of-the-art methods.

REFERENCES

[1] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” SIAM Journal on imaging sciences, vol. 2, no. 1, pp. 183–202, 2009.

[2] 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.

[3] L. Xu, Q. Yan, Y. Xia, and J. Jia, “Structure extraction from texture via relative total variation,” ACM Transactions on Graphics (TOG), vol. 31, no. 6, p. 139, 2012.

[4] J. Cheng, Y. Gao, B. Guo, and W. Zuo, “Image restoration using spatially variant hyper-laplacian prior,” Signal, Image and Video Processing, vol. 13, no. 1, pp. 155–162, 2019.

[5] D. Krishnan and R. Fergus, “Fast image deconvolution using hyper-laplacian priors,” in NeurIPS, 2009, pp. 1033–1041.

[6] R. Liu, G. Zhong, J. Cao, Z. Lin, S. Shan, and Z. Luo, “Learning to diffuse: A new perspective to design pdes for visual analysis,” IEEE TIP, vol. 27, no. 3, pp. 937–966, 2016.

[7] Y. Romano, M. Elad, and P. Milanfar, “The little engine that could: Regularization by denoising (red),” SIAM Journal on Imaging Sciences, vol. 10, no. 4, pp. 1804–1844, 2017.

[8] K. Zhang, W. Zuo, S. Gu, and L. Zhang, “Learning deep cnn denoiser prior for image restoration,” in IEEE CVPR, 2017, pp. 3929–3938.

[9] Y. Chen and T. Pock, “Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration,” IEEE TIPAM, vol. 39, no. 6, 2017.

[10] K. Gregor and Y. LeCun, “Learning fast approximations of sparse coding,” in ICMl. Omnipress, 2010, pp. 399–406.

[11] P. Mu, J. Chen, R. Liu, X. Fan, and Z. Luo, “Learning to diffuse: A new perspective to design pdes for visual analysis,” IEEE TIP, vol. 27, no. 12, pp. 2457–2471, 2016.

[12] Y. Romano, M. Elad, and P. Milanfar, “The little engine that could: Regularization by denoising (red),” SIAM Journal on Imaging Sciences, vol. 10, no. 4, pp. 1804–1844, 2017.

[13] K. Zhang, W. Zuo, S. Gu, and L. Zhang, “Learning deep cnn denoiser prior for image restoration,” in IEEE CVPR, 2017, pp. 3929–3938.

[14] Y. Chen and T. Pock, “Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration,” IEEE TIPAM, vol. 39, no. 6, 2017.

[15] K. Gregor and Y. LeCun, “Learning fast approximations of sparse coding,” in ICMl. Omnipress, 2010, pp. 399–406.

[16] P. Mu, J. Chen, R. Liu, X. Fan, and Z. Luo, “Learning to diffuse: A new perspective to design pdes for visual analysis,” IEEE TIP, vol. 27, no. 12, pp. 2457–2471, 2016.

[17] Y. Romano, M. Elad, and P. Milanfar, “The little engine that could: Regularization by denoising (red),” SIAM Journal on Imaging Sciences, vol. 10, no. 4, pp. 1804–1844, 2017.

[18] K. Zhang, W. Zuo, S. Gu, and L. Zhang, “Learning deep cnn denoiser prior for image restoration,” in IEEE CVPR, 2017, pp. 3929–3938.
[53] P. Arias, G. Facciolo, V. Caselles, and G. Sapiro, “A variational framework for exemplar-based image inpainting,” *IJCV*, vol. 93, no. 3, pp. 319–347, 2011.

[54] L. He and Y. Wang, “Iterative support detection-based split bregman method for wavelet frame-based image inpainting,” *IEEE TIP*, vol. 23, no. 12, 2014.

[55] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang, “Weighted nuclear norm minimization and its applications to low level vision,” *IJCV*, vol. 121, no. 2, pp. 183–208, 2017.

[56] S. Tariq, “Rain. nvidia whitepaper,” 2007.

[57] D. Martin, C. Fowlkes, D. Tal, and J. Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,” in *IEEE ICCV*, 2001, pp. 416–423.

[58] X. Fu, J. Huang, X. Ding, Y. Liao, and J. Paisley, “Clearing the skies: A deep network architecture for single-image rain removal,” *IEEE TIP*, vol. 26, no. 6, 2017.

[59] X. Fu, J. Huang, D. Zeng, Y. Huang, X. Ding, and J. Paisley, “Removing rain from single images via a deep detail network,” in *IEEE CVPR*, 2017.

[60] S. Gu, D. Meng, W. Zuo, and L. Zhang, “Joint convolutional analysis and synthesis sparse representation for single image layer separation,” in *IEEE ICCV*, 2017, pp. 1717–1725.

[61] W. Yang, R. T. Tan, J. Feng, J. Liu, Z. Guo, and S. Yan, “Deep joint rain detection and removal from a single image,” in *IEEE CVPR*, 2017.

[62] T.-X. Jiang, T.-Z. Huang, X.-L. Zhao, L.-J. Deng, and Y. Wang, “Fastderain: A novel video rain streak removal method using directional gradient priors,” *IEEE TIP*, 2018.

[63] H. Zhang and V. M. Patel, “Density-aware single image de-raining using a multi-stream dense network,” in *IEEE CVPR*, 2018, pp. 695–704.

**Risheng Liu** received the BSc and PhD degrees both in mathematics from the Dalian University of Technology in 2007 and 2012, respectively. He was a visiting scholar in the Robotic Institute of Carnegie Mellon University from 2010 to 2012. He served as Hong Kong Scholar Research Fellow at the Hong Kong Polytechnic University from 2016 to 2017. He is currently an associate professor with the Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Internal School of Information and Software Technology, Dalian University of Technology, Dalian, China. He is also with the Peng Cheng Laboratory, Shenzhen, China. His research interests include machine learning, optimization, computer vision and multimedia. He was a co-recipient of the IEEE ICME Best Student Paper Award in both 2014 and 2015. Two papers were also selected as Finalist of the Best Paper Award in ICME 2017. He is a member of the IEEE and ACM.

**Pan Mu** received the B.S. degree in Applied Mathematics from Henan University, China, in 2014, the M.S. degree in Operational Research and Cybernetics from Dalian University of Technology, China, in 2017. She is currently pursuing the PhD degree in Computational Mathematics at Dalian University of Technology, Dalian, China. She is with the Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian University of Technology, Dalian, China. Her research interests include computer vision, machine learning, optimization and control.

**Jian Chen** received the B.E. degree in Electronic Commerce from Dalian University of Technology, Dalian, China, in 2018. He is currently pursuing the master degree in software engineering at Dalian University of Technology, Dalian, China. He is with the Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian University of Technology, Dalian, China. His research interests include rain streaks removal, computer vision, deep learning.

**Zhongxuan Luo** received the B.S. degree in Computational Mathematics from Jilin University, China, in 1985, the M.S. degree in Computational Mathematics from Jilin University in 1988, and the PhD degree in Computational Mathematics from Dalian University of Technology, China, in 1991. He has been a full professor of the School of Mathematical Sciences at Dalian University of Technology since 1997. He is also with the Peng Cheng Laboratory, Shenzhen, China. His research interests include computational geometry and computer vision.

**Xin Fan** was born in 1977. He received the B.E. and Ph.D. degrees in information and communication engineering from Xian Jiaotong University, Xian, China, in 1998 and 2004, respectively. He was with Oklahoma State University, Stillwater, from 2006 to 2007, as a post-doctoral research Fellow. He joined the School of Software, Dalian University of Technology, Dalian, China, in 2009. He is currently a professor with the DUT-RU International School of Information Science & Engineering, Dalian University of Technology, Dalian, China. He is with the Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian, China. He is also with the Peng Cheng Laboratory, Shenzhen, China. His current research interests include computational geometry and machine learning, and their applications to low-level image processing and DTI-MR image analysis.