Preprint, IFB-Report No. 27 (05/2009), Institute of Mathematics and Scientific Computing, University of Graz.

AN EFFICIENT PRIMAL-DUAL METHOD FOR L^{1}TV IMAGE RESTORATION

YIQIU DONG*, MICHAEL HINTERMÜLLER†, AND MARRICK NERI°

Abstract. Image restoration based on an ℓ^1-data-fitting term and edge preserving total variation regularization is considered. The associated non-smooth energy minimization problem is handled by utilizing Fenchel-duality and dual regularization techniques. The latter guarantee uniqueness of the dual solution and an efficient way for reconstructing a primal solution, i.e. the restored image, from a dual solution. For solving the resulting primal-dual system, a semismooth Newton solver is proposed and its convergence is studied. The paper ends by a report on restoration results obtained by the new algorithm for salt-and-pepper or random-valued impulse noise including blurring. A comparison with other methods is provided as well.

1. Introduction

Consider an image u in a two-dimensional bounded domain Ω, which is blurred and corrupted by noise during acquisition and transmission. In order to restore u, variational models have had great success; see, e.g., [8, 9, 15, 16, 18, 22] as well as the monograph [23] and the many references therein. One of the best known and most influential examples is the total variation based model, which was proposed in [15] for removing Gaussian noise. Based on the Gaussian noise model, it combines an L^2-data-fitting term with total variation regularization, and relies on the energy minimization problem

$$\min_{u\in BV(\Omega)} \int_\Omega |u-d|^2 \, d\mathrm{x} + \alpha \int_\Omega |Du|, \tag{1.1}$$

where $d \in L^2(\Omega)$ is the degraded image, $\alpha > 0$ is the regularization parameter, $\mathrm{x} = (x, y)$, and $BV(\Omega)$ denotes the space of functions of bounded variation. We recall that a function $u \in L^1(\Omega)$ is in $BV(\Omega)$ if the BV-seminorm

$$\int_\Omega |Du| = \sup\left\{ \int_\Omega u \, \mathrm{div}\vec{v} \, d\mathrm{x} : \vec{v} \in (C_0^\infty(\Omega))^2, \|\vec{v}\|_\infty \le 1 \right\}$$

is finite [10]. Since the total variation term $\int_\Omega |Du|$ allows discontinuities in the restoration, it is able to preserve edges in the image effectively while still suppressing noise satisfactorily [21]. Concerning the efficient numerical solution of (1.1), recently primal-dual techniques were introduced and shown to be highly competitive; see, e.g., [4, 6, 12, 13].

Key words and phrases. Deblurring, duality, ℓ^1-data fitting, random-valued impulse noise, salt-and-pepper noise, semismooth Newton, total variation regularization.

This research was funded by the Austrian Ministry of Science and Research through START program Y305 "Interfaces and Free Boundaries" and the Austrian Science Fund FWF through SFB "Mathematical Optimization and Applications in Biomedical Sciences".

In [17, 18], total variation regularization is combined with an L^1-data-fitting term for removing impulse noise, i.e., instead of (1.1) one considers

$$\min_{u\in BV(\Omega)} \int_\Omega |u-d|\ d\mathrm{x} + \alpha \int_\Omega |Du|.$$

We call the resulting non-smooth energy the L^1TV-model. Analytical and numerical results show that the L^1TV-model has some interesting properties, such as multiscale decomposition, contrast preservation, and morphological invariance [5, 17, 25]. These lead to the successful application of the L^1TV-model in medical imaging [25] and computer vision [5].

In this paper, we focus on a rather general L^1TV-model for the simultaneous deblurring and denoising of images. The discrete model reads as follows:

$$(\mathcal{P}) \qquad \min_{u\in\mathbb{R}^n} \|Ku-d\|_1 + \alpha \sum_{k=1}^{n} |[\nabla u]_k|_2,$$

where $d\in\mathbb{R}^m$, with $m\le n$, is the degraded image obtained from a two-dimensional pixel-array by concatenation in the usual column-wise fashion, $K\in\mathbb{R}^{m\times n}$ is a known linear (blurring) operator, and n is the number of pixels in the original image. The case of $m<n$ occurs, for instance, when parts of the image are missing. Moreover, $\|\cdot\|_1$ denotes the ℓ^1-vector-norm, and the discrete gradient operator $\nabla\in\mathbb{R}^{2n\times n}$ is defined by

$$\nabla v = \begin{bmatrix} \nabla_x v \\ \nabla_y v \end{bmatrix},$$

for $v\in\mathbb{R}^n$ with $\nabla_x,\ \nabla_y\in\mathbb{R}^{n\times n}$ corresponding to the discrete derivative in the x-direction and y-direction, respectively. In addition, throughout the paper we shall frequently use the following notation: For a vector $v\in\mathbb{R}^n$, we write $(v)_k = v_k$ to identify the k-th component of v and we use $[\vec{w}]_k := (\vec{w}_k, \vec{w}_{n+k})^\top$, with $\vec{w}\in\mathbb{R}^{2n}$, as well as $|[\vec{w}]_k|_2 := \sqrt{|\vec{w}_k|^2 + |\vec{w}_{k+n}|^2}$. Note that the symbol '$\vec{\cdot}$' is used to distinguish vectors in $\mathbb{R}^{2n}$, which result from discretizing two-dimensional vector fields over Ω, from vectors in $\mathbb{R}^n$, which typically relate to the discrete image or the given discrete data d. Below, we refer to the discrete objective in $(\mathcal{P})$ as the L^1TV-model, as well.

As the ℓ^1-data-fitting term in $(\mathcal{P})$ is convex but not strictly convex, the solution of the L^1TV-model is not unique, in general. Moreover, both terms in the objective are non-differentiable, which poses additional algorithmic challenges. In order to overcome both difficulties, we utilize Fenchel-duality and introduce associated inexact semismooth Newton techniques to treat the non-smoothness, and we propose a specific dual regularization to address the non-uniqueness. The resulting solution algorithm converges locally at a super-linear rate. Comparing with the recently proposed efficient FTVd-method [24], which uses a quadratic penalty function technique to smooth the ℓ^1-data-fitting term and the total variation term, our algorithm outperforms in image restoration ability at a comparable CPU-time consumption. In addition, we also compare our algorithm with the BKS-method proposed in [1], which combines the ℓ^1-data-fitting term with the Mumford-Shah regularization. The numerical results show that our algorithm is significantly better in terms of computational efficiency.

The outline of this paper is as follows. In Section 2 we derive the Fenchel dual problem of $(\mathcal{P})$ and introduce a dual regularization. Section 3 describes our semismooth Newton

based algorithm in detail. Numerical results are shown in Section 4, which demonstrate the performance of our algorithm. Finally, conclusions are drawn in Section 5.

2. The Fenchel-Dual of the L[1]TV-Model and Its Regularization

In this section, we approach the non-uniqueness of the solution and the non-differentiablility of the objective in $(\mathcal{P})$, the primal problem, by a Fenchel duality technique. It turns out that once a solution of the Fenchel dual is obtained, one faces the issue of computing an associated primal solution. For this purpose a specific dual regularization is employed.

2.1. Fenchel-dual of the L[1]TV-model.

Applying the Fenchel-Legendre duality calculus as in, e.g., [13], we derive the Fenchel-dual of $(\mathcal{P})$. For this purpose, we define $\Psi : \mathbb{R}^n \to \mathbb{R}$ and $\Phi : \mathbb{R}^{2n} \to \mathbb{R}$ by

$$\begin{aligned}\Psi(u) &:= \|Ku - d\|_1,\\ \Phi(\vec{p}) &:= \alpha \sum_{k=1}^{n} |[\vec{p}]_k|_2, \qquad \vec{p} := \nabla u \in \mathbb{R}^{2n}.\end{aligned}$$

As in [13], we obtain the Fenchel conjugate $\Phi^* : \mathbb{R}^{2n} \to \mathbb{R}$ as

$$\Phi^*(\vec{p}^*) = \mathfrak{I}_{\{\vec{w}:|[\vec{w}]_k|_2 \le \alpha,\ k=1,\dots,n\}}(\vec{p}^*),$$

where $\mathfrak{I}_S$ is the indicator function of the set $S \subset \mathbb{R}^{2n}$, i.e., $\mathfrak{I}_S(v) = 0$ if $v \in S$; $\mathfrak{I}_S(v) = +\infty$, otherwise. In addition, we have the following result.

Proposition 2.1. *If $KK^\top$ is invertible, then the Fenchel conjugate $\Psi^* : \mathbb{R}^n \to \mathbb{R}$ of Ψ is given by*

$$\Psi^*(u^*) = \langle d,\ (KK^\top)^{-1}Ku^*\rangle + \mathfrak{I}_{\{v:|((KK^\top)^{-1}Kv)_k| \le 1,\ k=1,\dots,m\}}(u^*),$$

where the vector inner product $\langle u,\ v\rangle = u^\top v = \sum_{k=1}^m u_k v_k$ is used.

Proof. Based on the definition of the Fenchel conjugate according to [7], i.e., $\Psi^*(v^*) = \sup_{v\in\mathbb{R}^n}\{\langle v,\ v^*\rangle - \Psi(v)\}$, we infer

$$\Psi^*(u^*) = \sup_{u\in\mathbb{R}^n}\left\{\langle u,\ u^*\rangle - \|Ku - d\|_1\right\}. \tag{2.1}$$

We readily find that Ψ^* attains its maximum at

$$u^* = K^\top v, \quad \text{for some } v \in \partial\|Ku - d\|_1. \tag{2.2}$$

Since $Ku^* = KK^\top v$ and $KK^\top$ is invertible, we get

$$v = (KK^\top)^{-1}Ku^* \tag{2.3}$$

and further $|((KK^\top)^{-1}Ku^*)_k| \le 1$ for all $k = 1,\dots,m$. Utilizing (2.3) and (2.2) in (2.1) yields

$$\begin{aligned}\Psi^*(u^*) &= \langle u,\ K^\top v\rangle - \|Ku - d\|_1\\ &= \langle Ku,\ v\rangle - \langle Ku - d,\ v\rangle\\ &= \langle d,\ v\rangle\\ &= \langle d,\ (KK^\top)^{-1}Ku^*\rangle.\end{aligned}$$

□

Concerning the flexibility of our approach, the following remark is in order.

Remark 2.2. *In Proposition 2.1 we assume that $KK^\top$ is invertible. In fact, in most cases such as Gaussian or out-of-focus blur with Dirichlet or Neumann boundary conditions the blurring operator $K \in \mathbb{R}^{n\times n}$ is invertible but ill-conditioned. Thus, in these situations and in the same way as in the proof of Proposition 2.1, we readily obtain $\Psi^*(u^*) = \langle d,\ K^{-\top}u^*\rangle + \mathfrak{I}_{\{v:|(K^{-\top}v)_k|\le 1,\ k=1,\dots,n\}}(u^*)$ with $m = n$.*

In cases where $K \in \mathbb{R}^{n\times n}$ is singular, we may set $\tilde{K} = K + \varepsilon I_n$, where $I_n \in \mathbb{R}^{n\times n}$ is the identity matrix. Then, we consider

$$\begin{aligned}(\mathcal{P}) &= \min_{u\in\mathbb{R}^n} \|\tilde{K}u - d\|_1 + \alpha\sum_{k=1}^n |[\nabla u]_k|_2 + \|Ku - d\|_1 - \|\tilde{K}u - d\|_1\\ &\le \min_{u\in\mathbb{R}^n} \|\tilde{K}u - d\|_1 + \alpha\sum_{k=1}^n |[\nabla u]_k|_2 + \|Ku - \tilde{K}u\|_1\\ &= \min_{u\in\mathbb{R}^n} \|\tilde{K}u - d\|_1 + \alpha\sum_{k=1}^n |[\nabla u]_k|_2 + \varepsilon\|u\|_1.\end{aligned}$$

Hence, for $\varepsilon \to 0$ we may use $\min_{u\in\mathbb{R}^n} \|\tilde{K}u - d\|_1 + \alpha\sum_{k=1}^n |[\nabla u]_k|_2$ to approximate the original problem $(\mathcal{P})$.

Note, however, that in Proposition 2.1 we assume $KK^\top$ instead of K invertible. This makes our subsequent algorithm applicable for solving a more general problem class, where the blurring operator K can be in $\mathbb{R}^{m\times n}$ with $m \le n$. Whenever $m < n$, the degraded image d might not only have been blurred and corrupted by noise, but it is incomplete. In Section 4, we show some numerical examples for such cases.

Transferring the general results of [7, Chapter III.4 Page 60-61] to our situation, we find that no duality gap occurs, i.e.,

$$\inf_{u\in\mathbb{R}^n}\{\Psi(u) + \Phi(\Lambda u)\} = \sup_{\vec{p}\in\mathbb{R}^{2n}}\{-\Psi^*(\Lambda^\top\vec{p}) - \Phi^*(-\vec{p})\}, \tag{2.4}$$

with $\Lambda = \nabla$. The dual problem of $(\mathcal{P})$, i.e. the right hand side in (2.4), is equivalent to

$$\begin{aligned}&\min_{\vec{p}\in\mathbb{R}^{2n}} \langle d,\ (KK^\top)^{-1}K\nabla^\top\vec{p}\rangle,\\ &\text{subject to (s. t.) } |((KK^\top)^{-1}K\nabla^\top\vec{p})_k| \le 1,\ k = 1,\dots,m,\\ &\qquad\qquad |[\vec{p}]_k|_2 \le \alpha,\ k = 1,\dots,n.\end{aligned} \tag{$\mathcal{P}^*$}$$

By duality theory, $(\mathcal{P}^*)$ admits a solution. Due to the lack of strict convexity of the objective and, in addition, the non-triviality of the kernel of div $:= -\nabla^\top$, the solution is not unique, in general.

Based on the primal-dual system, a solution $(\bar{u}, \bar{\vec{p}})$ is characterized by

$$\begin{aligned}\Lambda^\top\bar{\vec{p}} &\in \partial\Psi(\bar{u}),\\ -\bar{\vec{p}} &\in \partial\Phi(\Lambda\bar{u}).\end{aligned} \tag{2.5}$$

Associated with $(\mathcal{P})$ and $(\mathcal{P}^*)$, the latter conditions are equivalent to

$$-\mathrm{div}\bar{\vec{p}} = K^\top \bar{v}, \quad \text{for some } \bar{v} \in \partial\|K\bar{u} - d\|_1, \tag{2.6a}$$

$$\begin{cases} |[\nabla \bar{u}]_k|_2 [\bar{\vec{p}}]_k + \alpha[\nabla \bar{u}]_k = 0, & \text{if } |[\bar{\vec{p}}]_k|_2 = \alpha, \\ [\nabla \bar{u}]_k = 0, & \text{if } |[\bar{\vec{p}}]_k|_2 < \alpha, \end{cases} \quad \text{for } k = 1, \dots, n. \tag{2.6b}$$

We observe that if $|((KK^\top)^{-1}K\nabla^\top \bar{\vec{p}})_k| < 1$, then we have $(K\bar{u})_k = d_k$. If $|((KK^\top)^{-1}K\nabla^\top \bar{\vec{p}})_k| = 1$, on the other hand, we are only able to characterize $(K\bar{u})_k$ as greater than or less than d_k, but we do not get an exact value for $K\bar{u}_k$. Such an incomplete characterization of $\bar{u}$ conforms with the nonuniqueness of solutions of $(\mathcal{P})$ due to the non-strict convexity of the objective.

2.2. **Regularization.** Next we modify the dual problem $(\mathcal{P}^*)$ in order to cope with the non-uniqueness of the dual solution and the difficulty of recovering a primal solution from a solution of $(\mathcal{P}^*)$. As a remedy to these two problematic situations, we propose to add two regularization terms to the objective of $(\mathcal{P}^*)$ yielding

$$\begin{aligned} (\mathcal{P}^*_{\lambda,\gamma}) \qquad & \min_{\vec{p}\in\mathbb{R}^{2n}} \langle d,\ (KK^\top)^{-1}K\nabla^\top \vec{p}\rangle + \frac{\lambda}{2}\|(KK^\top)^{-1}K\nabla^\top \vec{p}\|_2^2 + \frac{\gamma}{2\alpha}\|\vec{p}\|_2^2, \\ & \text{s. t. } |((KK^\top)^{-1}K\nabla^\top \vec{p})_k| \le 1,\ k = 1, \dots, m, \\ & \qquad |[\vec{p}]_k|_2 \le \alpha,\ k = 1, \dots, n. \end{aligned}$$

where $\|\cdot\|_2$ denotes the ℓ^2-vector-norm, $\gamma > 0$ and $\lambda > 0$ are the associated regularization parameters. Observe that the term involving γ yields uniqueness of the solution of $(\mathcal{P}^*_{\lambda,\gamma})$; compare [13]. In order to understand the impact of the λ-regularization, we study the dual of $(\mathcal{P}^*_{\lambda,\gamma})$ and relate the resulting problem to $(\mathcal{P})$. For this purpose we introduce the functions $\Psi^*_\lambda : \mathbb{R}^n \to \mathbb{R}$ and $\Phi^*_\gamma : \mathbb{R}^{2n} \to \mathbb{R}$ as

$$\begin{aligned} \Psi^*_\lambda(\nabla^\top \vec{p}) :=& \langle d,\ (KK^\top)^{-1}K\nabla^\top \vec{p}\rangle + \frac{\lambda}{2}\|(KK^\top)^{-1}K\nabla^\top \vec{p}\|^2 \\ & + \mathfrak{I}_{\{\vec{v}:|((KK^\top)^{-1}K\nabla^\top \vec{v})_k|\le 1,\ k=1,\cdots,m\}}(\vec{p}), \\ \Phi^*_\gamma(-\vec{p}) :=& \frac{\gamma}{2\alpha}\|\vec{p}\|_2^2 + \mathfrak{I}_{\{\vec{w}:|[\vec{w}]_k|_2\le\alpha,\ k=1,\cdots,n\}}(\vec{p}). \end{aligned}$$

With these definitions $(\mathcal{P}^*_{\lambda,\gamma})$ is equivalent to

$$\min_{\vec{p}\in\mathbb{R}^{2n}} \Psi^*_\lambda(\nabla^\top \vec{p}) + \Phi^*_\gamma(-\vec{p}).$$

Like in [13], based on the definition of the Fenchel conjugate we obtain the Fenchel dual of Φ^*_γ as

$$\Phi_\gamma(\vec{q}) = \frac{\alpha}{2\gamma} \sum_{\{k:|[\vec{q}]_k|_2<\gamma\}} |[\vec{q}]_k|_2^2 + \alpha \sum_{\{k:|[\vec{q}]_k|_2\ge\gamma\}} \left[|[\vec{q}]_k|_2 - \frac{\gamma}{2}\right]. \tag{2.7}$$

In addition, we have the following conclusion concerning the Fenchel dual of Ψ^*_λ.

Proposition 2.3. *The Fenchel dual of Ψ_λ^* is given by*

$$\Psi_\lambda(u) = \frac{1}{2\lambda} \sum_{\{k:|(Ku)_k - d_k|<\lambda\}} |(Ku)_k - d_k|^2 + \sum_{\{k:|(Ku)_k-d_k|\geq\lambda\}} \left[|(Ku)_k - d_k| - \frac{\lambda}{2} \right].$$

Proof. Based on the definition of the Fenchel conjugate, we have $\Psi_\lambda(u) = \sup_{u^*\in\mathbb{R}^n} \left\{ \langle u,\ u^*\rangle - \Psi_\lambda^*(u^*) \right\}$. Hence, Ψ_λ attains its maximum at

$$u = K^\top(KK^\top)^{-1}d + \lambda K^\top(KK^\top)^{-1}(KK^\top)^{-1}Ku^* + K^\top(KK^\top)^{-1}\mu, \tag{2.8}$$

where $\mu \in \mathbb{R}^n$ represents the multiplier associated with the constraint $|((KK^\top)^{-1}Ku^*)_k| \leq 1$ for all $k = 1, \ldots, m$ satisfying

$$\begin{aligned} \mu_k\big(((KK^\top)^{-1}Ku^*)_k - 1\big)\big(((KK^\top)^{-1}Ku^*)_k + 1\big) = 0, &\quad \forall k \in \{1,\ldots,m\}, \\ \mu_k \geq 0, &\quad \forall k \in \mathcal{A}^+, \\ \mu_k \leq 0, &\quad \forall k \in \mathcal{A}^-, \end{aligned}$$

where

$$\mathcal{A}^+ = \{k : ((KK^\top)^{-1}Ku^*)_k = 1\} \quad \text{and} \quad \mathcal{A}^- = \{k : ((KK^\top)^{-1}Ku^*)_k = -1\}.$$

Then, from (2.8) we obtain after multiplying by K (for the first relation) or taking the inner product with u^* (for the second relation)

$$\begin{aligned} 0 &= Ku - d - \lambda(KK^\top)^{-1}Ku^* - \mu, \\ 0 &= \langle u,\ u^*\rangle - \langle d,\ (KK^\top)^{-1}Ku^*\rangle - \lambda\|(KK^\top)^{-1}Ku^*\|_2^2 - \langle \mu,\ (KK^\top)^{-1}Ku^*\rangle. \end{aligned}$$

A slight rearrangement yields

$$(KK^\top)^{-1}Ku^* = \frac{1}{\lambda}(Ku - d - \mu), \tag{2.9a}$$

$$\langle u,\ u^*\rangle = \langle d,\ (KK^\top)^{-1}Ku^*\rangle + \lambda\|(KK^\top)^{-1}Ku^*\|^2 + \langle \mu,\ (KK^\top)^{-1}Ku^*\rangle. \tag{2.9b}$$

Employing (2.9a) and (2.9b) in the Fenchel dual of Ψ_λ^*, we obtain

$$\begin{aligned} \Psi_\lambda(u) &= \frac{\lambda}{2}\|(KK^\top)^{-1}Ku^*\|_2^2 + \langle \mu,\ (KK^\top)^{-1}Ku^*\rangle \\ &= \frac{1}{2\lambda}\|Ku - d - \mu\|_2^2 + \frac{1}{\lambda}\langle \mu,\ Ku - d - \mu\rangle \\ &= \frac{1}{2\lambda}\|Ku - d\|_2^2 - \frac{1}{2\lambda}\|\mu\|_2^2. \end{aligned}$$

In order to represent $\|\mu\|_2^2$ in terms of Ku, d and λ, observe that

(i) $\mu_k = (Ku)_k - d_k - \lambda \geq 0$ for all $k \in \mathcal{A}^+$;
(ii) $\mu_k = (Ku)_k - d_k + \lambda \leq 0$ for all $k \in \mathcal{A}^-$;
(iii) $\mu_k = 0$ for all $k \in \mathcal{I} = \{k : |((KK^\top)^{-1}Ku^*)_k| < 1\}$.

From this we get

$$\|\mu\|_2^2 = \sum_{k\in\mathcal{A}^+} |(Ku)_k - d_k - \lambda|^2 + \sum_{k\in\mathcal{A}^-} |(Ku)_k - d_k + \lambda|^2$$

and therefore

$$\begin{aligned}\Psi_\lambda(u) &= \frac{1}{2\lambda}\|Ku-d\|_2^2 - \frac{1}{2\lambda}\sum_{k\in\mathcal{A}^+}|(Ku)_k - d_k - \lambda|^2 - \frac{1}{2\lambda}\sum_{k\in\mathcal{A}^-}|(Ku)_k - d_k + \lambda|^2 \\ &= \frac{1}{2\lambda}\sum_{k\in\mathcal{I}}|(Ku)_k - d_k|^2 + \sum_{k\in\mathcal{A}^+}\left[(Ku)_k - d_k - \frac{\lambda}{2}\right] + \sum_{k\in\mathcal{A}^-}\left[d_k - (Ku)_k - \frac{\lambda}{2}\right] \\ &= \frac{1}{2\lambda}\sum_{k\in\mathcal{I}}|(Ku)_k - d_k|^2 + \sum_{k\in\mathcal{A}^+\cup\mathcal{A}^-}\left[|(Ku)_k - d_k| - \frac{\lambda}{2}\right].\end{aligned}$$

□

Remark 2.4. *Comparing Φ_γ with Φ in the original problem $(\mathcal{P})$, the total variation regularization is smoothed locally in order to obtain the uniqueness of the solution of $\vec{p}$ in the dual problem. In addition, as a remedy for the incomplete characterization when recovering a primal from a dual solution, from Proposition 2.3 we find that the ℓ^1-data-fitting term Ψ is smoothed locally yielding Ψ_λ. The latter now allows to retrieve a solution of the primal problem associated with $(\mathcal{P}^*_{\lambda,\gamma})$.*

According to (2.4), (2.7) and Proposition 2.3, the Fenchel dual problem of $(\mathcal{P}^*_{\lambda,\gamma})$ is given by

$$(\mathcal{P}_{\lambda,\gamma}) \qquad \min_{u\in\mathbb{R}^n} \Psi_\lambda(u) + \Phi_\gamma(\nabla u).$$

which approximates the original primal problem $(\mathcal{P})$. In order to clarify the precise relation between $(\mathcal{P}_{\lambda,\gamma})$ and $(\mathcal{P})$ we have the following theorem.

Theorem 2.5. *Suppose $\bar{u}_{\lambda,\gamma}$ is a solution of the minimization problem $(\mathcal{P}_{\lambda,\gamma})$ and that there exists a constant $\tilde{C}$ (independent of λ and γ) such that $\|\bar{u}_{\lambda,\gamma}\|_1 \le \tilde{C}$. Then, as $\lambda,\gamma \to 0$ every accumulation point of $\{\bar{u}_{\lambda,\gamma}\}$ is a solution of $(\mathcal{P})$.*

Proof. Let $\mathcal{J}(u)$ and $\mathcal{J}_{\lambda,\gamma}(u)$ denote the objective functions of $(\mathcal{P})$ and $(\mathcal{P}_{\lambda,\gamma})$, respectively, i.e.

$$\mathcal{J}(u) = \Psi(u) + \Phi(\nabla u), \quad \mathcal{J}_{\lambda,\gamma}(u) = \Psi_\lambda(u) + \Phi_\gamma(\nabla u).$$

Then, we have

$$\begin{aligned}|\mathcal{J}_{\lambda,\gamma}(u) - \mathcal{J}(u)| \le& |\Psi_\lambda(u) - \Psi(u)| + |\Phi_\gamma(u) - \Phi(u)| \\ \le& \frac{1}{2\lambda}\left|\sum_{\{k:|(Ku)_k - d_k|<\lambda\}}\left(|(Ku)_k - d_k|^2 - 2\lambda|(Ku)_k - d_k|\right)\right| \\ &+ \left|\sum_{\{k:|(Ku)_k - d_k|\ge\lambda\}}\left(|(Ku)_k - d_k| - \frac{\lambda}{2} - |(Ku)_k - d_k|\right)\right| \\ &+ \frac{\alpha}{2\gamma}\left|\sum_{\{k:|[\nabla u]_k|_2<\gamma\}}\left(|[\nabla u]_k|_2^2 - 2\gamma|[\nabla u]_k|_2\right)\right| \\ &+ \alpha\left|\sum_{\{k:|[\nabla u]_k|_2\ge\gamma\}}\left(|[\nabla u]_k|_2 - \frac{\gamma}{2} - |[\nabla u]_k|_2\right)\right|\end{aligned}$$

$$
\begin{aligned}
=&\frac{1}{2\lambda}\left|\sum_{\{k:|(Ku)_k-d_k|<\lambda\}}\left[\left(|(Ku)_k-d_k|-\lambda\right)^2-\lambda^2\right]\right|+\sum_{\{k:|(Ku)_k-d_k|\geq\lambda\}}\frac{\lambda}{2}\\
&+\frac{\alpha}{2\gamma}\left|\sum_{\{k:|[\nabla u]_k|_2<\gamma\}}\left[\left(|[\nabla u]_k|_2-\gamma\right)^2-\gamma^2\right]\right|+\alpha\sum_{\{k:|[\nabla u]_k|_2\geq\gamma\}}\frac{\gamma}{2}\\
\leq&\frac{1}{2}\left(\lambda m+\alpha\gamma n\right). \qquad (2.10)
\end{aligned}
$$

Further, for any $u_1\in\mathbb{R}^n$ and $u_2\in\mathbb{R}^n$, set

$$
\begin{aligned}
\mathcal{A}_1&:=\{k:|(Ku_1)_k-d_k|<\lambda,\ |(Ku_2)_k-d_k|<\lambda\},\\
\mathcal{A}_2&:=\{k:|(Ku_1)_k-d_k|<\lambda,\ |(Ku_2)_k-d_k|\geq\lambda\},\\
\mathcal{A}_3&:=\{k:|(Ku_1)_k-d_k|\geq\lambda,\ |(Ku_2)_k-d_k|<\lambda\},\\
\mathcal{A}_4&:=\{k:|(Ku_1)_k-d_k|\geq\lambda,\ |(Ku_2)_k-d_k|\geq\lambda\},
\end{aligned}
$$

and

$$
\begin{aligned}
\mathcal{A}_5&:=\{k:|[\nabla u_1]_k|_2<\gamma,\ |[\nabla u_2]_k|_2<\gamma\},\\
\mathcal{A}_6&:=\{k:|[\nabla u_1]_k|_2<\gamma,\ |[\nabla u_2]_k|_2\geq\gamma\},\\
\mathcal{A}_7&:=\{k:|[\nabla u_1]_k|_2\geq\gamma,\ |[\nabla u_2]_k|_2<\gamma\},\\
\mathcal{A}_8&:=\{k:|[\nabla u_1]_k|_2\geq\gamma,\ |[\nabla u_2]_k|_2\geq\gamma\}.
\end{aligned}
$$

Then we obtain

$$
\begin{aligned}
|\mathcal{J}_{\lambda,\gamma}(u_1)-\mathcal{J}_{\lambda,\gamma}(u_2)|\leq&|\Psi_\lambda(u_1)-\Psi_\lambda(u_2)|+|\Phi_\gamma(u_1)-\Phi_\gamma(u_2)|\\
\leq&\frac{1}{2\lambda}\left|\sum_{k\in\mathcal{A}_1}\left(|(Ku_1)_k-d_k|^2-|(Ku_2)_k-d_k|^2\right)\right|\\
&+\left|\sum_{k\in\mathcal{A}_2}\left[\frac{1}{2\lambda}|(Ku_1)_k-d_k|^2-\left(|(Ku_2)_k-d_k|-\frac{\lambda}{2}\right)\right]\right|\\
&+\left|\sum_{k\in\mathcal{A}_3}\left[\left(|(Ku_1)_k-d_k|-\frac{\lambda}{2}\right)-\frac{1}{2\lambda}|(Ku_2)_k-d_k|^2\right]\right|\\
&+\left|\sum_{k\in\mathcal{A}_4}\left(|(Ku_1)_k-d_k|-|(Ku_2)_k-d_k|\right)\right|\\
&+\frac{\alpha}{2\gamma}\left|\sum_{k\in\mathcal{A}_5}\left(|[\nabla u_1]_k|_2^2-|[\nabla u_2]_k|_2^2\right)\right|\\
&+\left|\sum_{k\in\mathcal{A}_6}\left[\frac{\alpha}{2\gamma}|[\nabla u_1]_k|_2^2-\alpha\left(|[\nabla u_2]_k|_2-\frac{\gamma}{2}\right)\right]\right|\\
&+\left|\sum_{k\in\mathcal{A}_7}\left[\alpha\left(|[\nabla u_1]_k|_2-\frac{\gamma}{2}\right)-\frac{\alpha}{2\gamma}|[\nabla u_2]_k|_2^2\right]\right|
\end{aligned}
$$

$$
\begin{aligned}
&+\alpha\Big|\sum_{k\in\mathcal{A}_8}\big(|[\nabla u_1]_k|_2-|[\nabla u_2]_k|_2\big)\Big|\\
\leq&\frac{1}{2\lambda}\sum_{k\in\mathcal{A}_1}\big(|(Ku_1)_k-d_k|+|(Ku_2)_k-d_k|\big)\big|(Ku_1)_k-(Ku_2)_k\big|\\
&+\sum_{k\in\mathcal{A}_2}\Big[|(Ku_2)_k-d_k|-\frac{1}{2\lambda}\big(|(Ku_1)_k-d_k|-\lambda\big)^2-|(Ku_1)_k-d_k|\Big]\\
&+\sum_{k\in\mathcal{A}_3}\Big[|(Ku_1)_k-d_k|-\frac{1}{2\lambda}\big(|(Ku_2)_k-d_k|-\lambda\big)^2-|(Ku_2)_k-d_k|\Big]\\
&+\sum_{k\in\mathcal{A}_4}\big|(Ku_1)_k-(Ku_2)_k\big|\\
&+\frac{\alpha}{2\gamma}\sum_{k\in\mathcal{A}_5}\big(|[\nabla u_1]_k|_2+|[\nabla u_2]_k|_2\big)\big||[\nabla u_1]_k|_2-|[\nabla u_2]_k|_2\big|\\
&+\alpha\sum_{k\in\mathcal{A}_6}\Big[|[\nabla u_2]_k|_2-\frac{1}{2\gamma}\big(|[\nabla u_1]_k|_2-\gamma\big)^2-|[\nabla u_1]_k|_2\Big]\\
&+\alpha\sum_{k\in\mathcal{A}_7}\Big[|[\nabla u_1]_k|_2-\frac{1}{2\gamma}\big(|[\nabla u_2]_k|_2-\gamma\big)^2-|[\nabla u_2]_k|_2\Big]\\
&+\alpha\sum_{k\in\mathcal{A}_8}\big||[\nabla u_1]_k|_2-|[\nabla u_2]_k|_2\big|\\
\leq&\sum_{k\in\mathcal{A}_1\cup\mathcal{A}_4}\big|(Ku_1)_k-(Ku_2)_k\big|\\
&+\sum_{k\in\mathcal{A}_2\cup\mathcal{A}_3}\big||(Ku_1)_k-d_k|-|(Ku_2)_k-d_k|\big|\\
&+\alpha\||\nabla u_1|_2-|\nabla u_2|_2\|_1\\
\leq&\|Ku_1-Ku_2\|_1+\alpha\||\nabla u_1|_2-|\nabla u_2|_2\|_1\\
\leq&\|K(u_1-u_2)\|_1+\alpha\||\nabla(u_1-u_2)|_2\|_1\\
\leq&C\|u_1-u_2\|_1.
\end{aligned}
$$

Since there exists $\tilde{C}$ such that $\|\bar{u}_{\lambda,\gamma}\|_1\leq\tilde{C}$, we can find a Cauchy sequence $\{\bar{u}_{\lambda_l,\gamma_l}\}$ of minimizers of $\mathcal{J}_{\lambda_l,\gamma_l}$, which converges to some u^* as $\lambda_l,\gamma_l\to 0$ with $l\to+\infty$. Since

$$
\begin{aligned}
|\mathcal{J}_{\lambda_l,\gamma_l}(\bar{u}_{\lambda_l,\gamma_l})-\mathcal{J}(u^*)|&=|\mathcal{J}_{\lambda_l,\gamma_l}(\bar{u}_{\lambda_l,\gamma_l})-\mathcal{J}_{\lambda_l,\gamma_l}(u^*)+\mathcal{J}_{\lambda_l,\gamma_l}(u^*)-\mathcal{J}(u^*)|\\
&\leq|\mathcal{J}_{\lambda_l,\gamma_l}(\bar{u}_{\lambda_l,\gamma_l})-\mathcal{J}_{\lambda_l,\gamma_l}(u^*)|+|\mathcal{J}_{\lambda_l,\gamma_l}(u^*)-\mathcal{J}(u^*)|,
\end{aligned}
$$

our previous considerations and $\bar{u}_{\lambda_l,\gamma_l}\to u^*$ as $l\to+\infty$ yield

$$
\lim_{l\to+\infty}\mathcal{J}_{\lambda_l,\gamma_l}(\bar{u}_{\lambda_l,\gamma_l})=\mathcal{J}(u^*).
$$

Suppose $\bar{u}$ is a minimizer of $\mathcal{J}(u)$, then taking into account the specific structure of Ψ_λ and Φ_γ we readily obtain

$$\mathcal{J}(\bar{u}) \leq \mathcal{J}(\bar{u}_{\lambda_l,\gamma_l}) \leq \mathcal{J}_{\lambda_l,\gamma_l}(\bar{u}_{\lambda_l,\gamma_l}) \leq \mathcal{J}_{\lambda_l,\gamma_l}(\bar{u}). \tag{2.11}$$

Making use of (2.10), we get

$$\lim_{l\to+\infty} \mathcal{J}_{\lambda_l,\gamma_l}(\bar{u}) = \mathcal{J}(\bar{u})$$

and hence

$$\lim_{l\to+\infty} \mathcal{J}_{\lambda_l,\gamma_l}(\bar{u}_{\lambda_l,\gamma_l}) = \mathcal{J}(\bar{u}).$$

The latter yields $\mathcal{J}(u^*) = \mathcal{J}(\bar{u})$. Since $\bar{u}$ is a minimizer of $\mathcal{J}(u)$, u^* is also a minimizer, i.e. $\{\bar{u}_{\lambda_l,\gamma_l}\}$ converges to a solution of $(\mathcal{P})$ as $\lambda_l, \gamma_l \to 0$. □

Remark 2.6. *Note that the objective function of our discrete problem $(\mathcal{P})$ is rescaled when compared to a discretization of its continuous counterpart. This is responsible for the dependence of the bound in (2.10) on m and n. Using some quadrature rules for discretizing $\int_\Omega |f|\,\mathrm{dx}$ yields, e.g., $h_1h_2\sum_{k=1}^n |f_k|$, when f_k is the value of f at the pixel x_k, $h_1 = \frac{1}{n_1}$ and $h_2 = \frac{1}{n_2}$ with n_1 and n_2 the number of pixels in x- and y-direction, respectively. Proceeding in such a way eliminates the dependence on $n = n_1n_2$ and on m in (2.10).*

Remark 2.7. *From the definition of $\mathcal{J}$, we have*

$$\mathcal{J}(\bar{u}_{\lambda,\gamma}) \geq \|K\bar{u}_{\lambda,\gamma} - d\|_1 \geq \|K\bar{u}_{\lambda,\gamma}\|_1 - \|d\|_1.$$

Assuming that there exists a constant $\hat{C} > 0$ (independent of u) such that $\|Ku\|_1 \geq \hat{C}\|u\|_1$ for $u \in \mathbb{R}^n$, based on (2.11) we get

$$\hat{C}\|\bar{u}_{\lambda,\gamma}\|_1 \leq \|d\|_1 + \mathcal{J}_{\lambda,\gamma}(\bar{u}) \leq C(\bar{u})(1 + \mathcal{J}(\bar{u})),$$

for $\lambda, \gamma \leq 1$ and a constant $C(\bar{u}) > 0$. Hence, the boundedness assumption of Theorem 2.5 is automatically satisfied. Furthermore, adding a few more technicalities, this assumption on K may be weakened to $\ker(K) \cap \ker(\nabla) = \{0\}$; compare, for instance, [24] for a similar condition.

Based on (2.5), the solutions $u_{\lambda,\gamma}$ and $\vec{p}_{\lambda,\gamma}$ of the regularized problems $(\mathcal{P}_{\lambda,\gamma})$ and $(\mathcal{P}^*_{\lambda,\gamma})$, respectively, satisfy

$$-\mathrm{div}\vec{p}_{\lambda,\gamma} = K^\top D(m^\lambda)^{-1}(Ku_{\lambda,\gamma} - d), \tag{2.12a}$$

$$-\vec{p}_{\lambda,\gamma} = \alpha D(\vec{m}^\gamma)^{-1}\nabla u_{\lambda,\gamma}, \tag{2.12b}$$

where $D(v) \in \mathbb{R}^{\tilde{n}\times\tilde{n}}$ is a diagonal matrix with the vector $v \in \mathbb{R}^{\tilde{n}}$ as the main diagonal, $(m^\lambda)_k = \max\{\lambda, |(Ku_{\lambda,\gamma})_k - d_k|\}$ with $m^\lambda \in \mathbb{R}^m$, and $(\vec{m}^\gamma)_k = (\vec{m}^\gamma)_{n+k} = \max\{\gamma, |[\nabla u_{\lambda,\gamma}]_k|_2\}$ with $\vec{m}^\gamma \in \mathbb{R}^{2n}$. For numerical purposes we reformulate the equations (2.12a) and (2.12b) by setting

$$v_{\lambda,\gamma} := D(m^\lambda)^{-1}(Ku_{\lambda,\gamma} - d) \quad \text{and} \quad \vec{q}_{\lambda,\gamma} := -\vec{p}_{\lambda,\gamma}.$$

Then (2.12) can be equivalently expressed as

$$-D(m^\lambda)v_{\lambda,\gamma} + Ku_{\lambda,\gamma} - d = 0, \tag{2.13a}$$

$$\mathrm{div}\vec{q}_{\lambda,\gamma} - K^\top v_{\lambda,\gamma} = 0, \tag{2.13b}$$

$$D(\vec{m}^\gamma)\vec{q}_{\lambda,\gamma} - \alpha\nabla u_{\lambda,\gamma} = 0. \tag{2.13c}$$

Notice that the system (2.13) is non-smooth due to the presence of the max-operators. However, it can be shown that (2.13) is semismooth in the sense of [11]. Hence, in the next section we employ the semismooth Newton technique of [12] for its solution.

3. Semismooth Newton Algorithm

Before we commence with the description of our algorithm, we recall the notions of generalized (or Newton) differentiability and semismoothness of a mapping $F : \mathbb{R}^{n_1} \to \mathbb{R}^{n_2}$, $n_1, n_2 \in \mathbb{N}$. According to [11], F is called generalized (Newton or slant) differentiable in an open set $U \subset \mathbb{R}^{n_1}$ if there exists $G_F : \mathbb{R}^{n_1} \to \mathbb{R}^{n_2 \times n_1}$ such that

$$\|F(z+s) - F(z) - G_F(z+s)s\| = o(\|s\|) \text{ as } \|s\| \to 0$$

for all $z \in U$. This definition coincides with an equivalent characterization of semismoothness of locally Lipschitz maps F in [19] and it allows to generalize Newton's method to semismooth mappings: Given $z_0 \in \mathbb{R}^{n_1}$ sufficiently close to a root z^* of F, set $l := 0$ and iterate until some stopping rule is satisfied:

$$z_{l+1} = z_l - G_F(z_l)^{-1} F(z_l), \quad l := l + 1.$$

Under the additional assumption of bounded invertibility of G_F in a neighborhood of z^* (containing z_0) it can be shown that the above iteration converges q-superlinearly to z^*; see, e.g., [11].

Since it can readily be shown that the max-terms in (2.13) are semismooth, one may utilize a generalized Newton step at some current approximation $(u_l, v_l, \vec{q}_l)$ of a (primal-dual) solution of (2.13). This results in

$$\begin{bmatrix} A_l & -D(m_l^\lambda) & 0 \\ 0 & -K^\top & -\nabla^\top \\ B_l & 0 & D(\vec{m}_l^\gamma) \end{bmatrix} \begin{bmatrix} \delta_u \\ \delta_v \\ \delta_q \end{bmatrix} = \begin{bmatrix} -Ku_l + d + D(m_l^\lambda)v_l \\ \nabla^\top \vec{q}_l + K^\top v_l \\ \alpha \nabla u_l - D(\vec{m}_l^\gamma)\vec{q}_l \end{bmatrix}, \tag{3.1}$$

where

$$A_l = \Big[I_n - D(v_l)\chi_{\mathcal{A}_l^\lambda} D(\operatorname{sign}(Ku_l - d)) \Big] K =: \Lambda_l K,$$
$$B_l = \Big[-\alpha I_{2n} + D(\vec{q}_l)\chi_{\mathcal{A}_l^\gamma} D(\vec{m}_l^\gamma)^{-1} N(\nabla u_l) \Big] \nabla =: -C_l \nabla.$$

Further, $\mathcal{A}_l^\lambda = \{k : |(Ku_l)_k - d_k| \geq \lambda\}$, $\mathcal{A}_l^\gamma = \{k, n+k : |[\nabla u_l]_k|_2 \geq \gamma\}$, and, for $\mathcal{S} \subset \{1, \ldots, m\}$, $\chi_{\mathcal{S}} \in \mathbb{R}^{m \times m}$ denotes a diagonal matrix with the diagonal entries $t_k = 1$ if $k \in \mathcal{S}$ and $t_k = 0$ else for $k = 1, \ldots, m$. In addition, with $\nabla u \in \mathbb{R}^{2n}$,

$$N(\nabla u) := \begin{bmatrix} D(\nabla_x u) & D(\nabla_y u) \\ D(\nabla_x u) & D(\nabla_y u) \end{bmatrix} \in \mathbb{R}^{2n \times 2n}.$$

Since $D(m_l^\lambda)$ and $D(\vec{m}_l^\gamma)$ are invertible, we compute δ_v and δ_q from the first and third equations in (3.1) and substitute the result into the second equation. This yields an equation in δ_u only, i.e.,

$$H_l \delta_u = f_l, \tag{3.2}$$

where

$$H_l = K^\top D(m_l^\lambda)^{-1} \Lambda_l K + \nabla^\top D(\vec{m}_l^\gamma)^{-1} C_l \nabla,$$

$$f_l = -K^\top D(m_l^\lambda)^{-1}(Ku_l - d) - \alpha\nabla^\top D(\vec{m}_l^\gamma)^{-1}\nabla u_l.$$

In general, the matrix H_l is not symmetric because C_l is not. However, it was shown in [13] that the matrix C_l is symmetric at the solution $(u_l, v_l, \vec{q}_l) = (\bar{u}, \bar{v}, \bar{\vec{q}})$, and C_l is positive semidefinite whenever

$$|[\vec{q}_l]_k|_2 \le \alpha, \qquad \text{for } k = 1, \dots, n. \tag{C1}$$

In addition, the constraint

$$|(v_l)_k| \le 1, \qquad \text{for } k = 1, \dots, m, \tag{C2}$$

is required for the feasibility of the dual variable; compare $(\mathcal{P}^*_{\lambda,\gamma})$. When both (C1) and (C2) are satisfied, then we obtain the following lemma.

Lemma 3.1. *Suppose the conditions in (C1) and (C2) hold. Then, for all $l \in \mathbb{N}$, the matrix H_l is positive semidefinite.*

Proof. In the proof of Lemma 3.3 in [13], under the constraint (C1) C_l was shown to be positive semidefinite. Considering that all elements of the diagonal matrix $D(\vec{m}_l^\gamma)$ are positive, the second term of the sum in H_l is positive semidefinite. Since $D(m_l^\lambda)$ is a diagonal positive definite matrix, we only need to show that Λ_l is positive semidefinite in order to obtain the conclusion of the lemma. Clearly, Λ_l is a diagonal matrix, and the k-th diagonal element is

$$(\Lambda_l)_k = 1 - (v_l)_k(t_\lambda)_k \operatorname{sign}((Ku_l)_k - d_k).$$

Here, $(t_\lambda)_k$, $k = 1, \dots, m$, correspond to the diagonal entries of $\chi_{\mathcal{A}_l}$. Since $|(v_l)_k| \le 1$ for every $k \in \{1, \dots, m\}$, it follows that $(\Lambda_l)_k \ge 0$ for each k. Thus, Λ_l is positive semidefinite. □

Whenever the conditions (C1) or (C2) are not satisfied, in our algorithm we apply suitable projections of $\vec{q}_l$ and v_l onto their respective feasible region. In fact, for indices k which violate (C1), we replace $[\vec{q}_l]_k$ by $\alpha \max\{\alpha,\ |[\vec{q}_l]_k|_2\}^{-1}[\vec{q}_l]_k$. Similarly, if the condition (C2) is not met at some index k, we replace $(v_l)_k$ by $\operatorname{sign}((v_l)_k)$. As a result, we obtain a modified system matrix which we denote by H_l^+. Based on Lemma 3.1, we have the following result.

Theorem 3.2. *The matrix H_l^+ is positive semidefinite.*

Since there is no guarantee that H_l or H_l^+ is positive definite, a solution δ_u in (3.2) may not exist or may not be unique in case of existence. As a remedy, we add a small multiple of the identity matrix to the system matrix, and then use the biconjugate gradient stabilized (BICGSTAB) algorithm [20] to solve the resulting linear system. The overall Newton-type algorithm is as follows.

Algorithm for $\mathbf{L^1}$TV-Model:

- **1:** Initialize $(u_0, v_0, \vec{q}_0) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^{2n}$ and set $l := 0$.
- **2:** Estimate the active sets, i.e., determine $\chi_{\mathcal{A}^\lambda_{l+1}}$ and $\chi_{\mathcal{A}^\gamma_{l+1}}$.
- **3:** Compute H_l^+ if condition (C1) or (C2) is not satisfied; otherwise, set $H_l^+ = H_l$. Solve $(H_l^+ + \kappa_l I_n)\delta_u = f_l$ for δ_u, where κ_l is a small positive value approaching zero as l increases.

(a) (b)

FIGURE 1. Original images. (a) "Cameraman", (b) "Bridge".

4: Use δ_u to update $u_{l+1} = u_l + \delta_u$, and calculate v_{l+1} and $\vec{q}_{l+1}$ from

$$v_{l+1} = D(m_l^\lambda)^{-1}(A_l\delta_u + Ku_l - d),$$
$$\vec{q}_{l+1} = D(\vec{m}_l^\gamma)^{-1}(\alpha\nabla u_l - B_l\delta_u).$$

5: Stop; or set $l := l + 1$ and go to step 2.

We end this section by studying the convergence properties of our algorithm.

Theorem 3.3. *Suppose that $(\bar{u}, \bar{\vec{p}})$ denotes a solution pair of $(\mathcal{P}_{\lambda,\gamma})$ and $(\mathcal{P}^*_{\lambda,\gamma})$. Assume that $u_l \to \bar{u}$, $\vec{p}_l \to \bar{\vec{p}}$, and $\kappa_l \to 0$ as $l \to \infty$. Then H_l^+ converges to H_l as $l \to +\infty$.*

Proof. Since the assumption $\vec{p}_l \to \bar{\vec{p}}$ implies that

$$|[\vec{p}_l]_k|_2 \to |[\bar{\vec{p}}]_k|_2 \le \alpha \quad \text{and} \quad |(\mathrm{div}\vec{p}_l)_k| \to |(\mathrm{div}\bar{\vec{p}})_k| \le 1,$$

we obtain that H_l^+ converges to H_l. In addition, as we also have $\kappa_l \to 0$, $H_l^+ + \kappa_l I_n$ converges to H_l, as well. □

Similar to Theorem 3.6 in [13], from the above theorem we obtain the following result.

Theorem 3.4. *The iterates $(u_l, v_l, \vec{q}_l)$ of our algorithm converge superlinearly to $(\bar{u}, \bar{v}, \bar{\vec{q}})$ provided that $(u_0, v_0, \vec{q}_0)$ is sufficiently close to $(\bar{u}, \bar{v}, \bar{\vec{q}})$.*

Before reporting on numerical results obtained by our method, we point out that our approach including the semismooth Newton solver can be readily transferred to the case where the total variation regularization term is generalized with respect to the underlying vector-norm, i.e., when we use $\alpha\sum_{i=1}^n |[\nabla u]_k|_s$ for $s \ge 1$ instead of just $\alpha\sum_{i=1}^n |[\nabla u]_k|_2$. Concerning the handling of the regularization term in the primal objective, for $s = 1$ we refer to [12] and for $s > 1$ to [13], both in the case of (1.1).

4. Numerical Results

In this section, we provide numerical results to study the behavior of our method with respect to its image restoration capability and its computational efficiency. For illustrations, the results for the 256-by-256, 8-bit gray-level images "Cameraman" and "Bridge", see Figure

(a) (b) (c) (d)

FIGURE 2. Results of different methods when restoring noisy image "Cameraman" (row 1) and "Bridge" (row 2) corrupted by salt-and-pepper noise with the noise level $r = 30\%$. (a) Noisy images, (b) TV-method (row 1: SNR=6.78 dB; row 2: SNR=5.34 dB), (c) FTVd-method (row 1: SNR=12.52 dB; row 2: SNR=10.80 dB), (d) Our method (row 1: SNR=12.61 dB; row 2: SNR=11.36 dB).

1, are presented. The quality of the restoration results are shown quantitatively by using the signal-to-noise ratio (SNR) defined as

$$\mathrm{SNR} := 10 \log_{10} \frac{\|\bar{u} - \mathbf{E}(\bar{u})\|_2^2}{\|\bar{u} - u\|_2^2} \ (dB),$$

where $\bar{u}$ and u denote the original image and the restored image respectively, and $\mathbf{E}(\bar{u})$ is the mean gray-level value of the original image.

In our numerics, the $n \times n$-matrices ∇_x and ∇_y are obtained by applying finite difference approximations for the derivatives with symmetric boundary conditions in the respective coordinate direction. The κ-sequence for step 3 of our algorithm is generated by $\kappa_{l+1} = 10\frac{\mathrm{res}_l}{\mathrm{res}_0}$, where res_l denotes the ℓ_2-residual of the primal-dual system (2.13) at $(u_l, v_l, \vec{q}_l)$. Although Theorem 3.4 is a local convergence result only, the new algorithm always converged for our simple initial choice $u_0 = v_0 = (1, \ldots, 1)^\top$ and $\vec{p}_0 = 1.0\text{e-}6 \cdot (1, \ldots, 1)^\top$. We stop our algorithm as soon as $\mathrm{res}_l \leq 1.0\text{e-}3 \cdot \mathrm{res}_1$ for the first time.

For the algorithmic comparison reported on below we use the method of [13] for minimizing the L^2TV-model in (1.1) with a stopping rule analogous to the one above. The FTVd-algorithm was obtained from the authors of [24] and the BKS-method was implemented according to the detailed description (including the stopping rule) in [2] and companion work like [3], for instance.

(a) (b) (c) (d)

FIGURE 3. Results of different methods when restoring noisy image "Cameraman" (row 1) and "Bridge" (row 2) corrupted by random-valued impulse noise with the noise level $r = 30\%$. (a) Noisy images, (b) TV-method (row 1: SNR=7.09 dB; row 2: SNR=5.84 dB), (c) FTVd-method (row 1: SNR=12.91 dB; row 2: SNR=10.81 dB), (d) Our method (row 1: SNR=12.86 dB; row 2: SNR=11.21 dB).

4.1. **Restoration of Noisy Images.** For the behavior of denoising, we compare our method with two other techniques: the total variation (TV) method [15], which uses an ℓ^2-data-fitting term and is solved by the algorithm in [13], and the FTVd-method [24]. The FTVd-approach was proposed recently as an efficient method for the simultaneous deblurring and denoising of images subject to impulsive noise. It is based on the L^1TV-model and it uses a quadratic penalty function technique to cope with the non-differentiability of the objective. Since it can also be used for noise removal only, we use it for comparison.

In Figure 2 and 3 we show the results for removing salt-and-pepper noise and random-valued impulse noise with a noise level of $r = 30\%$. Moreover, the SNR-values for the three methods are listed. For the TV-method, after many experiments with different regularization parameter values α, the restored images with the best SNR-values are presented here. For the FTVd-method, based on the suggestions in [24], we fix its parameters $\beta = 2^{10}$ and $\gamma = 2^{15}$, and only adjust μ to achieve the best possible SNR-values. In our method, both Ψ_λ and Φ_γ are in fact Huber-type functions [14], and the parameters λ and γ control the balance between ℓ^1- and ℓ^2-data fitting. The smaller the parameter values are, the more difficult the minimization problem $(\mathcal{P}_{\lambda,\gamma})$ becomes. Here, based on our numerical experiments we choose $\lambda = 0.001$ and $\gamma = 0.01$. The remaining parameter α corresponds to the regularization and controls the trade-off between a good fit of d and a smoothness requirement due to the total variation regularization. This suggests that small α is used for low noise level in order to

	Salt-and-Pepper		Random-Valued	
	"Cameraman"	"Bridge"	"Cameraman"	"Bridge"
FTVd	10.68s	9.55s	9.78s	9.15s
Ours	7.62s	7.42s	5.19s	4.36s

TABLE 1. CPU time in seconds for restoring noisy images.

preserve details with little smoothing, and large α is used for high noise level in order to remove noise considerably. We determine α experimentally.

Concerning the results obtained by the different methods, we observe that the TV-method adversely affects the image contrast. Moreover, significant over-smoothing takes place. It is due to the continuous ℓ^2-data-fitting term, which corresponds to the Gaussian noise model. Compared to the TV-model and as expected, the results of the methods based on the L^1TV-model are much better with respect to contrast and smoothing. Since the FTVd-method and our method both aim at minimizing the L^1TV-functional they obtain similar results. We note, however, that the restored images obtained by the FTVd-method appear "smoother" than those from our method. This can be attributed to the quadratic penalty function technique utilized in the FTVd-method, which smoothes the ℓ^1-data-fitting term and the total variation term. In contrast, our method suppresses the noise successfully while preserving more details; to this end observe the grass region of "Cameraman" and the trees in the background of "Bridge". Quantitatively we note that our method yields the largest SNR-ratios.

For the comparison of computational efficiency, in Table 4.1 the CPU-times consumed by the FTVd-method and our method are listed, respectively. All simulations are run in Matlab 7.5 (R2007b) on a PC equipped with P4 3.0GHz CPU and 3G RAM memory. Observe that, although the FTVd-method is very efficient, in the denoising case our method spends less CPU-time.

To illustrate the convergence behavior of our method for solving the equation (3.2), in Table 4.1 we report on the norm of the residual $\| \operatorname{res}_l \|_2$, the number of iteration steps instep_l in the BICGSTAB algorithm and κ_l for each iteration. We note that a maximum of 30 iterations in the BICGSTAB algorithm is allowed. From Table 4.1 we see that the residual is decreasing in most iterations. Sometimes the residual increases (as, for instance, for $l = 18$ when removing salt-and-pepper noise), which can be attributed to a too small κ-value and the resulting ill-conditioning of the system matrix in (3.2). Although after few steps the number of the inner iterations reaches the maximum of 30, the restored images (compare Figure 2 and 3 (d)) are still acceptable. Based on our numerical experiments, allowing more inner iterations yields no significant effect on the results.

4.2. **Restoration of Blurred Noisy Images.** In order to discuss the behavior of our method in the deblurring case, we compare its performance with two current methods: the FTVd- and the BKS-method [2, 24]. The BKS-method combines the ℓ^1-data-fitting term with the Mumford-Shah regularization for restoration.

	Salt-and-Pepper			Random-Valued		
l	$\| \text{res}_l \|_2$	instep_l	κ_l	$\| \text{res}_l \|_2$	instep_l	κ_l
1	76.0	15	10.0	260.5	15	10.0
2	222.8	30	2.97	225.2	29	10.2
3	181.1	30	8.70	212.2	30	8.79
4	159.3	30	7.07	191.1	30	8.29
5	142.7	30	6.22	167.8	30	7.46
6	117.7	30	5.57	140.2	30	6.55
7	84.7	30	4.60	109.2	30	5.48
8	54.8	30	3.31	77.5	30	4.26
9	29.4	30	2.14	48.4	30	3.03
10	15.6	30	1.15	26.1	30	1.89
11	10.6	30	0.61	14.5	30	1.02
12	6.59	30	0.42	10.4	30	0.57
13	4.28	30	0.26	6.01	30	0.41
14	2.49	30	0.17	3.13	30	0.23
15	1.76	30	0.097	2.94	30	0.12
16	2.14	30	0.069	1.14	30	0.11
17	0.56	30	0.084	0.57	30	0.045
18	7.48	30	0.022	0.40	30	0.022
19	2.14	30	0.29	0.20	30	0.016
20	0.83	30	0.083			
21	0.15	30	0.033			

TABLE 2. Residual, the number of inner iterations and κ for each iteration l, when restoring the image "Bridge" with 30% impulse noise.

In Figure 4 and 5 we show the results for restoring the degraded images blurred by a Gaussian operator (with window size 7 and standard deviation 5) and corrupted by salt-and-pepper noise with different noise levels. Figure 6 shows the restoration results under the same blurring operator but corrupted by random-valued impulse noise. Furthermore, we also provide the SNR-values for the three methods. Concerning the parameter choices, for the FTVd-method we follow the suggestions in [24]. The BKS-method requires the adjustment of three parameters; see [2]. We select these parameters one by one through numerical tests until they become stable. In our method, as we mentioned in the last section, we fix $\lambda = 0.001$, $\gamma = 0.01$ and determine α experimentally.

From Figure 4–6 we find that our method performs best both visually and quantitatively. Note that the images restored by the FTVd-method are "smoother" than the results from the other two methods. This might be due to the quadratic penalty function technique incorporated in FTVd. In addition, because the FTVd-method relies on periodic boundary conditions, which might not be adequate in all cases of real images but which is important for the diagonalization of the system matrix, it leads to artifacts near the boundary of the images. Although, compared with the FTVd-method, more details are preserved by the BKS-method, ringing artifacts are noticeable near the edges; see, e.g., the tripod-region in Figure 4(c). Since total variation regularization is utilized in our method, the spurious rings are absent from our results. Moreover, the usage of the Huber functions Ψ_λ and Φ_γ helps to preserve details in the images.

(a) (b) (c) (d)

FIGURE 4. Results of different methods when restoring blurred noisy image "Cameraman" corrupted by salt-and-pepper noise with different noise level (row 1: $r = 30\%$; row 2: $r = 50\%$). (a) Blurred noisy images, (b) FTVd-method (row 1: SNR=13.69 dB; row 2: SNR=12.33 dB), (c) BKS-method (row 1: SNR=15.20 dB; row 2: SNR=12.79 dB), (d) Our method (row 1: SNR=16.27 dB; row 2: SNR=13.50 dB).

noise		"Cameraman"			"Bridge"		
		FTVd	BKS	Ours	FTVd	BKS	Ours
SP	30%	12.95	1166.42	17.57	15.22	888.13	13.51
	50%	11.83	1983.51	16.27	12.64	1468.40	13.10
RV	30%	15.08	1207.20	19.87	13.61	967.66	15.63

TABLE 3. CPU-time in seconds for restoring blurred noisy images with different noise level. (SP: Salt-and-pepper noise; RV: Random-valued impulse noise)

Table 4.2 reports on the CPU-times consumed by the three methods in our comparison. Although the FTVd-method spends less CPU-time in most cases, our method is still comparable. However, since the FTVd-method is sped up by utilizing the quadratic penalty function technique, which smoothes the ℓ^1-data-fitting term and the total variation regularization, and the periodic boundary condition that helps to diagonalize the system matrix, its restoration results appear over-smoothed compared to the results due to our method. Also, the FTVd-based results include some artifacts near the image boundary; see Figure

(a) (b) (c) (d)

FIGURE 5. Results of different methods when restoring blurred noisy image "Bridge" corrupted by salt-and-pepper noise with different noise level (row 1: $r = 30\%$; row 2: $r = 50\%$). (a) Blurred noisy images, (b) FTVd-method (row 1: SNR=10.41 dB; row 2: SNR=9.37 dB), (c) BKS-method (row 1: SNR=12.98 dB; row 2: SNR=10.86 dB), (d) Our method (row 1: SNR=13.09 dB; row 2: SNR=11.18 dB).

4–6. This indicates that our method performs better than the FTVd-method with a similar CPU-time requirement. Compared with the BKS-method, in our tests we find that our method is much faster. Especially for restoring the blurred image with 50% salt-and-pepper noise, the BKS-method takes around 112-times more CPU-time than our method. Furthermore, in our method only the total variation regularization parameter needs to be determined rather than the selection of three parameters by various numerical tests in the BKS-method.

In Table 4.2 we list $\| \text{res}_l \|_2$, the number of inner iterations and κ_l corresponding to each iteration for restoring blurred noisy images. We find that the residual keeps decreasing and κ goes to zero.

4.3. **Restoration of Incomplete Blurred Noisy Images.** In this subsection, we consider the problem, where the image "Cameraman" is not only blurred and corrupted by impulse noise, but also loses some information after blurring. Hence, the blurring operator $K \in \mathbb{R}^{m \times n}$ is not a square matrix, but it satisfies $m < n$ as we mentioned in Remark 2.2. If $KK^\top$ is invertible, we can still use our method for recovery.

In Figure 7 we show the results for restoring the incomplete blurred noisy image by our method. In this example, the image "Cameraman" is blurred by a Gaussian operator with window size 7 and standard deviation 5. During the blurring 3 or 7 pixel-columns get lost;

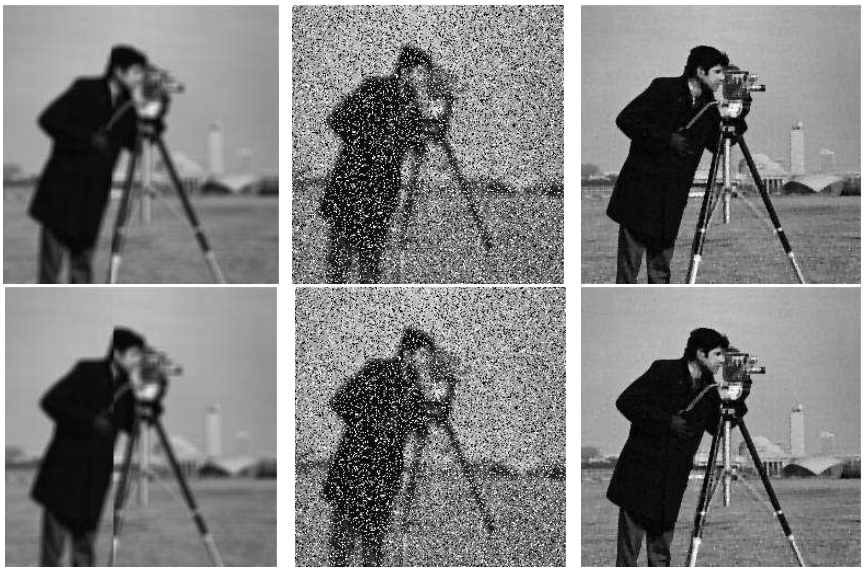
(a) (b) (c) (d)

FIGURE 6. Results of different methods when restoring blurred noisy image "Cameraman" and "Bridge" corrupted by random-valued impulse noise with the noise level $r = 30\%$. (a) Blurred noisy images, (b) FTVd-method (row 1: SNR=13.36 dB; row 2: SNR=10.41 dB), (c) BKS-method (row 1: SNR=14.77 dB; row 2: SNR=12.68 dB), (d) Our method (row 1: SNR=15.94 dB; row 2: SNR=12.90 dB).

	Salt-and-Pepper			Random-Valued		
l	$\|\| \text{res}_l \|\|_2$	instep_l	κ_l	$\|\| \text{res}_l \|\|_2$	instep_l	κ_l
1	42.1	8	10.0	260.9	3	10.0
2	214.8	14	1.64	226.7	5	10.2
3	156.8	6	8.39	211.5	7	8.85
4	146.8	11	6.12	195.0	9	8.26
5	113.4	14	5.73	175.6	10	7.62
6	71.9	18	4.43	148.9	13	6.86
7	38.1	25	2.81	125.6	15	5.81
8	20.6	30	1.49	92.2	19	4.91
9	7.12	30	0.81	56.1	24	3.60
10	2.50	30	0.28	25.7	30	2.19
11	1.18	30	0.098	10.8	30	1.00
12	0.65	30	0.046	3.25	30	0.42
13	0.39	30	0.026	1.32	30	0.13
14	0.19	30	0.015	0.69	30	0.051
15				0.39	30	0.027
16				0.21	30	0.015

TABLE 4. Residual, the number of inner iterations and κ for each iteration l, when restoring the blurred image "Bridge" with 30% impulse noise.

compare Figure 7(a). After these degradation operations, the images are also corrupted



(a) (b) (c)

FIGURE 7. Results of our method when restoring incomplete blurred noisy image "Cameraman" corrupted by salt-and-pepper noise with the noise level $r = 30\%$. (a) Incomplete blurred images (row 1: 3-column missing; row 2: 7-column missing), (b) Incomplete blurred noisy image, (c) Results by our method.

by salt-and-pepper noise with a noise level of 30%; see Figure 7(b). From the restoration results obtained by our method shown in Figure 7(c), we find that the missing image part is recovered successfully during deblurring and denoising. This holds true even in the case of a loss of 7 columns, which is the same size as the window size of the blurring operator.

5. Conclusion

A Fenchel-duality based approach for minimizing the L^1TV-functional for restoring blurred images subject to random-valued impulse or salt-and-pepper noise is proposed. The associated non-smooth data-fitting term causes problems when recovering a primal solution, i.e. the restored image, from a solution of the Fenchel dual. This difficulty can be overcome by adding a dual regularization involving the blurring operator. Further, the non-differentiability due to the total variation regularization results in simple norm-constraints in the dual problem. Due to the gradient-operator in the TV-regularization, the solution of the dual is in general non-unique. Uniqueness is obtained by adding a simple quadratic regularization to the dual objective. Both of the dual regularizations result in a local smoothing of the respective term in the primal problem such that the corresponding first order optimality (or Euler-Lagrange) system turns out to be non-differentiable but semismooth. The latter observation allows to employ semismooth Newton techniques for solving the system and to prove their locally superlinear rate of convergence. Compared to other recently proposed

methods, our approach appears to be very competitive with respect to image restoration capabilities and CPU-time consumption.

Acknowledgement. The authors would like to thank Dr. Wotao Yin (Rice University) for making his code implementing the method in [24] available and for discussions.

References

[1] L. Bar, N. Sochen, and N. Kiryati. Image deblurring in the presence of salt-and-pepper noise. In *Proceeding of 5th International Conference on Scale Space and PDE methods in Computer Vision*, volume LNCS, pages 107–118, 2005.

[2] L. Bar, N. Sochen, and N. Kiryati. Image deblurring in the presence of impulsive noise. *International Journal of Computer Vision*, 70:279–298, 2006.

[3] L. Bar, N. Sochen, and N. Kiryati. Deblurring of color images corrupted by salt-and-pepper noise. *IEEE Transactions on Image Processing*, 16:1101–1111, 2007.

[4] A. Chambolle. An algorithm for total variation minimization and application. *Journal of Mathematical Imaging and Vision*, 20:89–97, 2004.

[5] T. F. Chan and S. Esedoglu. Aspects of total variation regularized L^1 function approximation. *SIAM J. Appl. Math.*, 65:1817–1837, 2005.

[6] T. F. Chan, G. H. Golub, and P. Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM J. Sci. Comput.*, 20(6):1964–1977 (electronic), 1999.

[7] I. Ekeland and R. Témam. *Convex Analysis and Variational Problems*. Classics Appl. Math. 28, SIAM, 1999.

[8] H. Fu, M. K. Ng, M. Nikolova, and J. L. Barlow. Efficient minimization methods of mixed $\ell 2 - \ell 1$ and $\ell 1 - \ell 1$ norms for image restoration. *SIAM J. Sci. Comput.*, 27:1881–1902, 2006.

[9] G. Gilboa, N. Sochen and Y.Y. Zeevi. Texture preserving variational denoising using an adaptive fidelity term. *In Proc. of IEEE Workshop on Variational, Geometric and Level Set Methods in Computer Vision, Nice, France*, pages 137–144, 2003.

[10] E. Giusti. *Minimal Surfaces and Functions of Bounded Variation*. Monogr. Math. 80, Birkhäuser-Verlag, 1984.

[11] M. Hintermüller, K. Ito, and K. Kunisch. The primal-dual active set strategy as a semismooth newton method. *SIAM J. Optim.*, 13:865–888, 2002.

[12] M. Hintermüller and K. Kunisch. Total bounded variation regularization as bilaterally constrained optimization problem. *SIAM J. Appl. Math.*, 64:1311–1333, 2004.

[13] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.

[14] P. J. Huber. Robust regression: Asymptotics, conjectures, and monte carlo. *Annals of Statistics*, 1:799–821, 1973.

[15] L. I. Rudin, S. Osher and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[16] D. Mumford and J. Shah. Optimal approximation by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.*, 17:577–685, 1989.

[17] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers. *SIAM Journal on Numerical Analysis*, 40:965–994, 2002.

[18] M. Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20:99–120, 2004.

[19] L. Q. Qi and J. Sun. A nonsmooth version of Newton's method. *Math. Program.*, 58:353–367, 1993.

[20] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, 2 edition, 2003.

[21] D. Strong and T. Chan. Edge-preserving and scale-dependent properties of total variation regularization. *Inverse Problems*, 19:165–187, 2003.

[22] A. Tikhonov and V. Arsenin. *Solutions of Ill-Posed Problems*. Winston and Sons, Washington D.C, 1977.

[23] C. R. Vogel. *Computational Methods for Inverse Problems*. Frontiers Appl. Math. 23, SIAM Philadelphia, 2002.

[24] J. Yang, Y. Zhang, and W. Yin. An efficient TVL1 algorithm for deblurring multichannel images corrupted by impulsive noise. Rice University, CAAM Technical Report TR08-12.
[25] W. Yin, D. Goldfarb, and S. Osher. The total variation regularized L^1 model for multiscale decomposition. *SIAM Journal on Multiscale Modeling and Simulation*, 6:190–211, 2006.

* Institute of Mathematics and Scientific Computing, University of Graz, Heinrichstrasse 36, A-8010 Graz, Austria.
E-mail address: yiqiu.dong@uni-graz.at

† Department of Mathematics, Humboldt-University of Berlin, Unter den Linden 6, 10099 Berlin, Germany.
E-mail address: hint@math.hu-berlin.de

° University of the Philippines-Diliman, Quezon City 1101, Philippines.
E-mail address: marrick@math.upd.edu.ph