Article

Accelerated Chambolle Projection Algorithms for Image Restoration

Wenyang Wei † and Xiangchu Feng *,†

School of Mathematics and Statistics, Xidian University, Xi’an 710071, China
* Correspondence: xcfeng@mail.xidian.edu.cn
† These authors contributed equally to this work.

Abstract: In this paper, the accelerated Chambolle projection algorithms based on Frank–Wolfe have been proposed. For solving the image restoration under the additive Gaussian noise, the Chambolle projection method (CP) is widely used. However, the projection operator has a large computational cost and complex form. By means of the Frank–Wolfe method, this projection operation can be greatly simplified. We propose two new algorithms, called Chambolle projection based on Frank–Wolfe (CP–FW) and Chambolle projection based on accelerated Frank–Wolfe (CP–AFW). They have a fast convergence rate and low computation cost. Furthermore, we extend the new algorithms to deal with the Poisson noise. The convergence of the new algorithms is discussed, and results of the experiment show their effectiveness and efficiency.

Keywords: Chambolle projection; accelerated Frank–Wolfe method; Gaussian noise; Poisson noise

1. Introduction

In image processing, minimizing the total variation energy functional is popular [1–4]. Recently, many variants of the total variation energy functional model, the fractional total variation [5], the high-order total variation [6], directional total variation [7], and total generalized variation [8] have been applied for denoising, deblurring, inpainting, and other tasks. For solving the minimizing total variation, plenty of scholars have studied algorithms, for example, the standard [9] or non-standard discretization scheme [10], operator splitting [11–13], the Chambolle–Pock dual algorithm [11,14], the alternating direction method [15–17], and the Chambolle projection [4].

For solving the image restoration under the additive Gaussian noise, the Chambolle projection method (CP) is widely used. However, the projection operator has a large computational cost and complex form. By means of the Frank–Wolfe method, this projection operation can be greatly simplified.

In this paper, we propose new accelerated Chambolle projection algorithms for Gaussian denoising, according to the work of Chambolle. We call our new methods Chambolle projection based on Frank–Wolfe (CP–FW) and Chambolle projection based on accelerated Frank–Wolfe (CP–AFW). Meanwhile, we generalize the new algorithms under the Poisson noise condition. The convergence of algorithms is proved. Experiments show the better convergence rate of new algorithms and a low computational cost.

The outline of this paper is organized as follows. In Section 2, we show the related works and declare the notation. In Section 3, new algorithms are proposed under the Gaussian noise condition and a theoretical analysis is given. In Section 4, we extend these algorithms for Poisson noise removal. Section 5 shows the experiments of new algorithms. In Section 6, we summarize this paper.
2. Related Works

In this section, we first review the Chambolle projection algorithms for minimizing the total variation under the Gaussian noise. Then, the Frank–Wolfe method is briefly discussed.

2.1. Chambolle Projection Algorithm

For $f = u + n$, minimizing total variation [18] under the Gaussian noise has

$$
\min_u E(u) = \frac{\|f - u\|_2^2}{2\lambda} + \int_\Omega |\nabla u|,
$$

where $f$ is the observed image, $u$ is the clean image, $n$ is the additive white Gaussian noise (AWGN), $\Omega$ is the image domain, and $\lambda$ is a regularization parameter to keep trade-off between data fidelity term and regularization term. $\|\cdot\|_2$ denotes the $L^2$ norm (for convenience, we rewrite $\|\cdot\|_2^2$ as $\|\cdot\|_2^2$ in this paper). $\int_\Omega |\nabla u|$ is a semi-norm in the bounded variation (BV) space.

For convenience, let $J(u) = \int_\Omega |\nabla u|$. Then, the Euler–Lagrange equation for (1) is shown as follows:

$$
0 \in \frac{1}{\lambda}(u - f) + \partial J(u),
$$

where $\partial$ is a sub-differential operator of $J(u)$. Define $w \in \partial J(u)$; then, (2) can be written as

$$
\frac{f - u}{\lambda} \in \partial J(u) \Leftrightarrow u \in \partial J^\ast \left( \frac{f - u}{\lambda} \right),
$$

where $J^\ast$ is the conjugate function of $J$. Then,

$$
0 \in \frac{f - u}{\lambda} - f + \frac{1}{\lambda} \partial J^\ast \left( \frac{f - u}{\lambda} \right).
$$

We obtain $w = \frac{f - u}{\lambda}$ is the minimizer of

$$
\min_w \frac{\|w - \left( \frac{f}{\lambda} \right)\|_2^2}{2} + \frac{J^\ast(w)}{\lambda}.
$$

(5)

Since $J^\ast$ is the indicator function $\chi_K(w)$, where $K$ is the closure of the set

$$
\left\{ \text{div} p \in C^1_c(\Omega, R^2), |p(x)|_\infty \leq 1, \forall x \in \Omega \right\},
$$

where $C^1_c(\Omega, R^2)$ represents the space of first-order continuously differentiable functions with compact support in domain $(\Omega, R^2)$. The divergence $\text{div}$ is defined by $\text{div} = -\nabla^\ast$ ($\nabla^\ast$ is the adjoint of $\nabla$). Thus, (5) becomes

$$
\min_{w \in K} \frac{\|w - \left( \frac{f}{\lambda} \right)\|_2^2}{2}.
$$

(6)

The objective function in (6) is a quadratic function, and the constraint set $K$ is convex. We deduce the solution $w = \pi_K \left( \frac{f}{\lambda} \right)$, where $\pi_K$ is a projection operator, and the solution of (1) is shown by

$$
u = f - \lambda \pi_K \left( \frac{f}{\lambda} \right).
$$

(7)
Next, we declare the notations in discrete setting. To simplify, we assume that the image $f$ is given by a two-dimensional matrix of size $N \times N$, although it would be easy to adapt all arguments for a more general $M \times N$ matrix to use the notations of chambole, $X = R^{N \times N}$, and $Y = X \times X$. If $u \in X$, the gradient $\nabla u \in Y$ is given by $(\nabla u)_{ij} = \left((\partial_x u)_{ij}, (\partial_y u)_{ij}\right)$ with
\[
(\partial_x u)_{ij} = \begin{cases} 
    u_{i+1,j} - u_{ij} & \text{if } i < N, \\
    0 & \text{if } i = N,
\end{cases}
\]
and
\[
(\partial_y u)_{ij} = \begin{cases} 
    u_{ij+1} - u_{ij} & \text{if } j < N, \\
    0 & \text{if } j = N,
\end{cases}
\]
for $i, j = 1, \cdots, N$. Then, $|\nabla u|_{ij} = \sqrt{(\partial_x u)_{ij}^2 + (\partial_y u)_{ij}^2}$ and $\int |\nabla u| = \sum_{1 \leq i, j \leq N} |\nabla u|_{ij}$. For $\|\cdot\|$, it is defined by $\|u\|^2 = \langle u, u \rangle_X$ and $\langle u, v \rangle_X = \sum_{ij} u(i,j)v(i,j)$. In the constraint set $K$, equivalently, for every $p \in Y$ and $u \in X$, $(-\text{div} p, u)_X = \langle p, \nabla u \rangle_Y$. In $Y$, we use the Euclidean scalar product, defined in the standard way by $\langle p, q \rangle_Y = \sum_{1 \leq i, j \leq N} (p^1, q^1) + (p^2, q^2)$ for every $p = (p^1, p^2), q = (q^1, q^2) \in Y$. It is easy to check that, for every $p = (p^1, p^2) \in Y$, discrete divergence $\text{div}$ is given by
\[
(\text{div} p)_{ij} = \begin{cases} 
    p^1_{ij} - p^1_{i-1,j} & \text{if } 1 < i < N, \\
    p^1_{ij} & \text{if } i = 1, \\
    -p^1_{i-1,j} & \text{if } i = N,
\end{cases} + \begin{cases} 
    p^2_{ij} - p^2_{ij-1} & \text{if } 1 < j < N, \\
    p^2_{ij} & \text{if } j = 1, \\
    -p^2_{ij-1} & \text{if } j = N.
\end{cases}
\]
Thus, the constraint set $K$ in discrete setting is the closure of
\[
\{ \text{div} p \in Y, |p_{ij}|_\infty \leq 1, \forall i, j \in Y \}.
\]

Chambolle suggests to solve the nonlinear projection $\pi_K$ of (6) by the following constraint minimization problem:
\[
\begin{align*}
\min_p & \quad \|\lambda \text{div} p - f\|_X^2 \\
\text{s.t.} & \quad p \in Y, |p_{ij}|_\infty \leq 1, \forall i, j = 1, \ldots, N,
\end{align*}
\]
and using the Lagrange multipliers and the fixed-point theory. Especially, let $\alpha_{ij} \geq 0$ be the Lagrange multipliers associated with each constraint in (9). We have for each $i, j$,$$
-(\nabla (\lambda \text{div} p - f))_{ij} + \alpha_{ij} p_{ij} = 0,$$
with either $\alpha_{ij} > 0$ and $|p_{ij}|_1 = 1$, or $|p_{ij}|_1 \leq 1$ and $\alpha_{ij} = 0$. Thus, we have that in any case
\[
\alpha_{ij} = \left| (\nabla (\lambda \text{div} p - f))_{ij} \right|.
\]
From this observation, applying the fixed-point algorithm (semi-implicit gradient descent), we compute $p_{ij}^{n+1}$ as
\[
p_{ij}^{n+1} = p_{ij}^n + \tau \left( (\nabla (\lambda \text{div} p - f))_{ij} - \left| (\nabla (\lambda \text{div} p - f))_{ij} \right| p_{ij}^{n+1} \right).
\]
Thus, a sequence of iterations for solving (9) are written as

\[ p_{n+1}^{i,j} = p_{n}^{i,j} + \frac{\tau (\nabla (\text{div} p^n - f / \lambda))_{i,j}}{1 + \tau |(\nabla (\text{div} p^n - f / \lambda))_{i,j}|}, \]

where \( \tau \) is a time step.

The Chambolle projection algorithm is shown as in Algorithm 1.

**Algorithm 1** Chambolle projection algorithm (CP)

**Input:** \( f, \lambda, \tau \)

**Initialization:**

1. \( u^0 = f, p^0 = 0 \)

2. \( \text{for } k = 1 : M \text{ do} \)
   1. \( p_{k+1}^{i,j} = p_{k}^{i,j} + \frac{\tau (\nabla (\text{div} p^k - f / \lambda))_{i,j}}{1 + \tau |(\nabla (\text{div} p^k - f / \lambda))_{i,j}|} \)
   2. \( \text{end for} \)

3. \( u = f - \lambda \text{div} p_{M+1} \)

4. **return** \( u \)

5. **Output:** The restoration clean image \( u \).

**Proposition 1** ([4]). Let \( \tau \leq \frac{1}{8}, \lambda \text{div} p^n \) is convergent to \( \pi_{\lambda K}(f) \) as \( n \to \infty \).

It is worth noting that the Chambolle projection applying semi-implicit gradient descent method to calculate the nonlinear projection problem (6). It has a convergence rate \( O\left(\frac{1}{k}\right)\).

2.2. Frank–Wolfe Algorithm

We consider the following constraint optimization problem:

\[ \min_{x \in \Theta} F(x), \]

where \( F \) is a smooth convex function and \( \Theta \) is a non-empty convex and compact constraint set. The Frank–Wolfe method, also known as the conditional gradient method, to solve optimization problem (11), is shown as Algorithm 2.

**Algorithm 2** Frank–Wolfe Algorithm (FW)

**Initialization:** \( x^0 \in \Theta, r_k = \frac{2}{k+2} \)

1. \( \text{for } k = 1 : n \text{ do} \)
   1. \( t^{k+1} = \text{arg min}_{t \in \Theta} \langle \nabla F(x^k), t \rangle \)
   2. \( x^{k+1} = (1 - r_k)x^k + r_k t^{k+1} \)
   3. \( \text{end for} \)

4. **return** \( x^n \)

Notice that the core idea is that the target function in Step 2 is a linear approximation of the original target function \( F(x) \), which is called the linear minimization oracle (LMO). Assuming that we apply the Frank–Wolfe method to (6), what will happen? By the linear minimization oracle (LMO), we will solve a linear approximation of the target function on the convex set \( K \), which is easy by the special construction of the set \( K \). The details are given in the next section.

3. Chambolle Projection Algorithms Based on Frank–Wolfe

Equation (6) is

\[ \min_{w} \left\| w - \left( \frac{\lambda}{\epsilon} \right) \right\|^2, s.t. w \in K. \]

(12)
Let \( H = \frac{\|w - f/\lambda\|^2}{2} \), then \( DH = (w - f/\lambda) \), where \( D \) is the functional derivative operator. By the Algorithm 2, we have

\[
y^{k+1} \in \arg\min_{y \in K} \langle DH(w^k), y \rangle = \arg\min_{y \in K} \langle w^k - f/\lambda, y \rangle.
\]

(13)

It is worth noting that (13) is a linear function and the set \( K \) in (8) has a special construction; thus, (13) is

\[
\arg\min_{y \in K} \langle w^k - f/\lambda, y \rangle = \arg\min_{\|p_{ij}\|_1 \leq 1} \sum_{ij} \langle \nabla(w^k - f/\lambda)_{ij}, p_{ij} \rangle.
\]

(14)

Therefore,

\[
(p^k)_{ij} \in \partial \| \nabla(w^k - f/\lambda)_{ij} \|_1 = \partial \| \left( \partial_x (w^k - f/\lambda)_{ij}, \partial_y (w^k - f/\lambda)_{ij} \right) \|_1,
\]

where \( \partial \|z\|_1 = sgn(z) \). \( sgn(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases} \). If \( z \) is vector or matrix, \( sgn(z) \) acts for each components of \( z \). Let

\[
p^k = sgn \left( \nabla \left( w^k - f/\lambda \right) \right),
\]

(15)

Obviously, \( y^k = div p^k \). To sum up, the solver of (12) can be expressed as follows:

\[
\begin{cases}
p^k = sgn \left( \nabla \left( w^k - f/\lambda \right) \right) \\
y^k = div p^k \\
w^{k+1} = (1 - r_s)w^k + r_s y^k.
\end{cases}
\]

(16)

Compared to (9), simple expressions and a lower computational cost are advantageous. Thus, we have the following Algorithm 3 about the Frank–Wolfe for improving the Chambolle projection to remove the Gaussian noise.

Algorithm 3 Chambolle projection based on Frank–Wolfe (CP–FW)

Input: \( f, \lambda \)

Initialization: \( u^0 = f, p^0 = 0, y^0 = 0, w^0 = 0, r_s = \frac{2}{s+2} \).

1: for \( s = 1 : n \) do
2: \( p^k = sgn \left( \nabla \left( w^k - f/\lambda \right) \right) \).
3: \( y^k = div p^k \).
4: \( w^{k+1} = (1 - r_s)w^k + r_s y^k \).
5: end for
6: \( u = f - \lambda w^{n+1} \).
7: return \( u \).
8: Output: The restoration clean image \( u \).

Proposition 2 ([19,20]). For (12), the Frank–Wolfe algorithm satisfies \( H(w^k) - H(w^*) \leq O \left( \frac{1}{k} \right) \), where \( w^* \) is the true solution of objective function \( H(w) \).
It means that CP–FW and CP have the same convergence rate under the worst condition. However, for the same objective function $H(w)$ and the constraint set $K$, CP–FW has simpler expressions than the expressions of Chambolle projection, and it can be further accelerated by the Nesterov momentum as follows in Algorithm 4, which is named as the Chambolle projection based on accelerated Frank–Wolfe (CP–AFW).

**Algorithm 4 Chambolle projection based on accelerated Frank–Wolfe (CP–AFW)**

**Input:** $f, \lambda$

**Initialization:** $u^0 = f, w^0 = 0, p^0 = 0, y^0 = 0, \theta^0 = 0, r = \frac{2}{k+3}$.

1: for $s = 1 : n$

2: $h^s = (1 - r_s)w^s + r_s y^s$.

3: $\theta^{s+1} = (1 - r_s)\theta^s + r_s (w^s - f) / \lambda$.

4: $p^{s+1} = \text{sgn}(\nabla(\theta^{s+1}))$.

5: $y^{s+1} = \text{div} p^{s+1}$.

6: $w^{s+1} = (1 - r_s)w^s + r_s y^{s+1}$.

7: end for

8: $u = f - \lambda w^n + 1$.

9: return $u$.

10: **Output:** The restoration clean image $u$.

**Proposition 3 ([21]).** For the optimization problem (12), the objective function $H$ is convex and has a Lipschitz continuous gradient, and the constraint set $K$ is convex and compact with diameter $D$. Further, $K$ is active, that is, $\|DH(w^*)\|_2^2 \geq G > 0$, where $G$ is constant. The constraint set $K$ is transformed as $L^p$ norm ball constraint set by (14). Especially, when $p = 2$, choosing $r_k = \frac{2}{k+3}$ and $\theta^0 = 0$, AFW guarantees acceleration with the convergence rate

$$H(w^k) - H(w^*) = O\left(\min\left\{\frac{LD^2T + C\ln k}{k^2}, \frac{L^2D}{k}\right\}\right), \forall k,$$

where $L$ is the Lipschitz continuous gradient constant, and $C$ and $T$ are the constants depending on $L, D$, and $G$. $w^*$ is the true solution of objective function $H(w)$.

Proposition 3 indicates the AFW at least has $O\left(\frac{1}{k}\right)$ convergence rate. The Nesterov momentum is still helpful for accelerating the FW algorithm.

Next, we extend our new algorithms to the Poisson noise removal.

## 4. Accelerated Chambolle Projection for Poisson Denoising

The Poisson noise can be represented mathematically as follows:

$$f = \text{Poiss}(u)$$

where $\text{Poiss}$ is the Poisson distribution. For Poisson noise removal, the classical methods include PCA [22], non-local means [23,24], Anscombe variance stabilizing transform [25–27], all kinds of variant total variation [11,15,28–30], and deep learning methods [31,32], using the convolution neural network.

It is noticeable that the total variation along the lines of the ROF model can also be used to the Poisson denoising problem. The total variation model for Poisson noise removal [9] is below:

$$\min_u E(u) = \int_\Omega \frac{(u - f\log u)}{\lambda} + \int_\Omega |\nabla u|,$$

where $\lambda$ is a regularization parameter and $\nabla$ is a gradient operator. We define $J(u) = \int_\Omega |\nabla u|$ again.
Especially, (18) has the existence and uniqueness of solution.

**Proposition 4** ([9]). $E(u)$ has a unique minimizer.

By the Euler–Lagrange equation for (18), we have

$$0 \in \frac{(u - f)}{\lambda u} + \partial f(u).$$

(19)

Please note that the derivation process of Chambolle projection under the Poisson noise condition is different from the Chambolle projection under the Gaussian noise condition. Equation (19) can be written as

$$\left(\frac{f - w}{\lambda w}\right) \in \partial f(w) \iff w \in \partial f^\ast \left(\frac{f - w}{\lambda w}\right).$$

(20)

We let $w = \frac{f - u}{\lambda u}$, then, we have $u = f - \lambda uw$. Thus, $\frac{f - w}{\lambda w} - w \in \partial f^\ast(w)$. Given by fixed $u$, we obtain the minimizing of the following optimization problem:

$$\min_w \left\| \frac{w - \left(\frac{f}{\lambda u}\right)}{2} \right\|^2 + f^\ast(w).$$

(21)

For (21), it is a projection process that we attain $w = \pi_{\lambda K} \left(\frac{f}{\lambda u}\right)$, and then we put $w = \pi_{\lambda K} \left(\frac{f}{\lambda u}\right)$ into $u = \frac{f}{1 + \lambda \pi_{\lambda K}(\frac{f}{\lambda u})}$. Thus, the solution of (18) can be calculated as

$$u = \frac{f}{1 + \lambda \pi_{\lambda K}(\frac{f}{\lambda u})},$$

(22)

(22) is an implicit equation of $u$; then, we have

$$\frac{f}{u^{k+1}} = 1 + \lambda \pi_{\lambda K}(\frac{f}{u^k}).$$

(23)

by the iteration method.

**Proposition 5** ([33]). Nonlinear projection $\pi_{\lambda K}$ is a firmly non-expansive operator.

**Theorem 1.** The sequence $\{u^k\}$ of (23) is convergent, i.e, $u^k \to u^*, k \to \infty$.

**Proof.** $\pi_{\lambda K}$ is a firmly non-expansive operator. For $\frac{f}{u^k} = 1 + \pi_{\lambda K}(\frac{f}{u^k})$, there exists $u^*$ satisfying $\frac{f}{u^*} = 1 + \pi_{\lambda K}(\frac{f}{u^*})$ as $k \to \infty$ by the fixed point theory. Noting that $u > 0$, thus $u^k \to u^*$ as $k \to \infty$. □

Next, we discuss the computation of the projection operator $\pi_{\lambda K}$. For each iteration, i.e, for fixed $k$, we adopt (24) to approximate $\pi_{\lambda K}(\frac{f}{\lambda u^k})$.

$$p_{i,j}^{n+1} = p_{i,j}^n + \tau \left(\nabla \left(\frac{u^k \text{div} p^n - f}{\lambda u}\right)\right)_{i,j} / (1 + \tau \left| \nabla \left(\frac{u^k \text{div} p^n - f}{\lambda u}\right)\right|)_{i,j}. \quad (24)$$

When we denote $f/u^k$ as $\hat{f}^k$, Theorem 2 theoretically describes the convergence of the method.

**Theorem 2.** Let $\tau \leq \frac{1}{\delta}$, $\lambda \text{div} p^n$ is convergent to $\pi_{\lambda K}(\hat{f}^k)$ as $n \to \infty$.

**Proof.** The proof is similar to the Proposition 1. See [4]. □
Under the Poisson noise condition, the Chambolle projection algorithm is a two-level loop algorithm. For improving Algorithm 5, we replace the inner loop with the Frank–Wolfe.

**Algorithm 5** Chambolle projection algorithm (CP (Poisson))

**Input:** \( f, \lambda, \tau, tol = 1 \times 10^{-3} \).

**Initialization:** \( u^0 = f, p^0 = 0, \delta^0 = 1 \).

1. while \( \delta^k \geq tol \) do
2.     for \( s = 1 : n \) do
3.         \( p^s_{ij}^k = \frac{p^s_{ij}^k (\nabla (u^k \div p^k) - f / \lambda))}{1 + \tau \| (\nabla (u^k \div p^k) - f / \lambda)) \|_2} \).
4.     end for
5.     \( u^{k+1} = \frac{f}{1 + \lambda \div p^{n+1}} \).
6.     \( \delta^{k+1} = \frac{\| u^{k+1} - u^k \|^2}{\| u^k \|^2} \) and let \( k = k + 1 \).
7. end while
8. return \( u \).

9. **Output:** The restoration clean image \( u \).

Rethinking the projection process of (21), it is

\[
\min_{w \in K} \frac{\| w - (f / \mu) \|^2}{2},
\]

and then update \( u \) by \( u = \frac{f}{1 + \lambda u} \). For solving (25) by Frank–Wolfe, let \( G = \frac{\| w - f / \lambda u \|^2}{2} \), then \( DG(w) = (w - f / \lambda u) \). We have

\[
\begin{align*}
    p^k &= sgn \left( \nabla \left( w^k - f / \lambda u \right) \right), \\
    y^k &= \div p^k, \\
    w^{k+1} &= (1 - r_k) w^k + r_k y^k.
\end{align*}
\]

Thus, we obtained the Chambolle projection based on the Frank–Wolfe for Poisson noise removal in Algorithm 6.

**Algorithm 6** Chambolle projection based on Frank–Wolfe for Poisson noise (CP–FW (Poisson))

**Input:** \( f, \lambda, tol = 1 \times 10^{-3} \).

**Initialization:** \( u^0 = f, p^0 = 0, y^0 = 0, w^0 = 0, \delta^0 = 1, r_s = \frac{2}{\pi^2} \).

1. while \( \delta^k \geq tol \) do
2.     for \( s = 1 : n \) do
3.         \( p^s = sgn \left( \nabla \left( w^s - f / \lambda u^s \right) \right) \).
4.         \( y^s = \div p^s \).
5.         \( w^{s+1} = (1 - r_s) w^s + r_s y^s \).
6.     end for
7.     \( u^{k+1} = \frac{f}{1 + \lambda u^{n+1}} \).
8.     \( \delta^{k+1} = \frac{\| u^{k+1} - u^k \|^2}{\| u^k \|^2} \) and let \( k = k + 1 \).
9. end while
10. return \( u \).
11. **Output:** The restoration clean image \( u \).
Meanwhile, the Chambolle projection algorithm based on the accelerated Frank–Wolfe for Poisson denoising can further accelerate the algorithm. See in Algorithm 7.

**Algorithm 7** Chambolle projection based on Accelerated Frank–Wolfe for Poisson noise (CP–AFW (Poisson))

**Input:** \( f, \lambda, \text{tol} = 1 \times 10^{-3} \).

**Initialization:** \( u^0 = f, w^0 = 0, p^0 = 0, y^0 = 0, \theta^0 = 0, r_s = \frac{2}{s+3} \).

**Output:** output result restoration clean image \( u \).

1: while \( \delta^k \geq \text{tol} \) do
2: \hspace{1em} for \( s = 1 : n \) do
3: \hspace{2em} \( h^s = (1 - r_s)w^s + r_s y^s \).
4: \hspace{2em} \( \theta^{s+1} = (1 - r_s)\theta^s + r_s \left( h^s - f / \lambda u_k \right) \).
5: \hspace{2em} \( p^{s+1} = \text{sgn} \left( \nabla \left( \theta^{s+1} \right) \right) \).
6: \hspace{2em} \( y^{s+1} = \text{div} p^{s+1} \).
7: \hspace{2em} \( w^{s+1} = (1 - r_s)w^s + r_s y^{s+1} \).
8: \hspace{1em} end for
9: \hspace{1em} \( u^{k+1} = \frac{f}{1 + \lambda u_{n+1}} \).
10: \hspace{1em} \( \delta^{k+1} = \frac{\|u^{k+1} - u_k\|^2}{\|u^k\|^2} \) and let \( k = k + 1 \).
11: end while
12: return \( u \).
13: **Output:** The restoration clean image \( u \).

**Theorem 3.** CP (Poisson), CP–FW (Poisson), and CP–AFW (Poisson) are convergent.

**Proof.** According to Theorems 1 and 2, the Chambolle projection for Poisson denoising converges. According to Theorem 1 and Proposition 2, the Chambolle projection based on the Frank–Wolfe algorithm for Poisson denoising converges. According to Theorem 1 and Proposition 3, the Chambolle projection based on the accelerated Frank–Wolfe for Poisson denoising converges.

Theoretical analysis indicates our simplified formulas are convergent. In the next section, numerical experiments show the effectiveness of our new algorithms.

5. **Experiments**

For the Chambolle projection and the accelerated Chambolle projection algorithms, we first show the change trend of sequences \( p^k \) of Algorithm 1 and \( w^k \) of Algorithms 3 and 4, or sequences \( p^k \) of Algorithm 5 and \( w^k \) of Algorithms 6 and 7 under the different noise conditions. Next, we test algorithms performance on the small dataset.

5.1. **Change Rate of Sequence and Value of Energy Functional**

Gaussian denoising: Firstly, we concentrate on the change rate of sequences \( p^k \) of Algorithm 1 and \( w^k \) of Algorithms 3 and 4 for Gaussian denoising. The “Papav” image is tested. The change rate of sequence is shown in Figure 1.

From Figure 1, we observe that the change rate of sequences \( p^k \) of Algorithm 1 and \( w^k \) of Algorithms 3 and 4: CP–AFW > CP–FW > CP, where horizontal axes represent iteration numbers and vertical axes represent the change rate of sequences (for a sequence of \( p^k \) of Algorithm 1 and \( w^k \) of Algorithms 3 and 4, relative error \( \varepsilon^{k+1} = \frac{\|p^{k+1} - p^k\|^2}{\|p^k\|^2} \) and \( \varepsilon^{k+1} = \frac{\|w^{k+1} - w^k\|^2}{\|w^k\|^2} \), and then observe the change trend of sequences). In the experiments, optimal iterations of CP are above 100 iterations, CP–FW is about 60–80 iterations, and CP–AFW is about 20–30 iterations. Therefore, under the theoretical proof of algo-
algorithm convergence, we verify accelerated Chambolle projection convergent faster than the Chambolle projection algorithm.

Figure 1. The change rate of sequences under the Gaussian noise condition.

Poisson denoising: Now, we consider the change rate of sequences $p^k$ of Algorithm 5 and $w^k$ of Algorithms 6 and 7 and the change of value of $E(u)$ energy functional with iterations of three algorithms for Poisson denoising, where the “Papav” image is tested. It can be verified in Figure 2.

From the Figure 2a, we can observe the same phenomenon of change trend like the Gaussian noise condition, where the horizontal axes represent inner loop iterations numbers and the vertical axes represent the change rate of sequences (for a sequence of $p^k$ of Algorithm 5 and $w^k$ of Algorithms 6 and 7, relative error $\varepsilon^{k+1} = \frac{\|p^{k+1} - p^k\|^2}{\|p^k\|^2}$ and $\varepsilon^{k+1} = \frac{\|w^{k+1} - w^k\|^2}{\|w^k\|^2}$, and then observe the change trend of sequences). From the Figure 2b, the value of $E(u)$ energy functional decreases with iterations and then remains almost unchanged, where horizontal axes represent outer-loop iterations numbers and vertical axes represent the value of energy functional. The value of the $E(u)$ energy functional of CP–FW is the smallest, followed by CP–AFW, and finally by CP. Note that the tip: the value of energy functional of Equation (18) can be calculated by $E(u) = \sum_{i,j} \left( (e^{u_i} - f_i)_{i,j} + \lambda |\nabla e^{u_i}|_{i,j} \right)$.

Meanwhile, by the experiments, the optimal inner loop iterations of CP are above 120 iterations, CP–FW is about 50–80 iterations, and CP–AFW is about 20–30 iterations. The outer loop iterations are about 3 iterations by the experiments.

Conclusion: Whether the Gaussian noise condition or Poisson, new accelerated Chambolle projection algorithms have simple expressions, a lower computational cost, and a faster convergence rate.
Figure 2. The change rate of sequences and the change of value of energy functional under the Poisson noise condition. (a) The change rate of sequences; (b) the change of the value of $E(u)$ energy function with iterations.

In order to highlight a lower computation cost of the accelerated Chambolle projection, we test one iteration running time on a large image “Airplane”, with a size of 512 × 512. The result can be seen in Figure 3a.

One iteration running time of CP–FW is the smallest. The running time of CP–AFW is similar to CP, but the formula of algorithms of CP–FW and CP–AFW is simpler than CP. Meanwhile, CP–AFW converges faster than CP. Thus, CP–FW and CP–AFW are superior to CP in the total computation cost.
5.2. Test Algorithms on Dataset

Next, we make a small image dataset for testing all of the algorithms for Gaussian denoising and Poisson denoising in image processing. The dataset includes 10 images, with a size of 256 × 256 pixels. We compare algorithms from the indexes of running time, PSNR and SSIM.

The small image dataset is shown as Figure 4.

Figure 3. Running time. (a) One iteration running time of three algorithms; (b) airplane.

Figure 4. Image dataset.
We apply algorithms for Gaussian denoising and Poisson denoising to the dataset. The experimental data can be seen in Figures 5 and 6. It is worth noting that the average PSNR of the dataset with the Gaussian noise is 22.2906 dB and the average SSIM is 0.3560. The average PSNR of the dataset with the Poisson noise is 26.4548 dB and the average SSIM is 0.5035.

![Figure 5](image1.png)  ![Figure 6](image2.png)

Figure 5. Algorithms of comparison for Gaussian denoising. (a) Running time; (b) the average PSNR of dataset for Gaussian denoising; and (c) the average SSIM of dataset for Gaussian denoising.

Figure 6. Algorithms of comparison for Poisson denoising. (a) Running time; (b) the average PSNR of dataset for Poisson denoising; and (c) the average SSIM of dataset for Poisson denoising.

From Figures 5 and 6, the relationship of the running time of all algorithms is shown: CP > CP–FW > CP–AFW. That is, CP has the longest running time on the dataset. The relationship of the PSNR of the algorithms is shown: CP > CP–FW > CP–AFW, and the relationship of SSIM of algorithms is also shown: CP > CP–FW > CP–AFW. Meanwhile, we observe that the index of the PSNR value is similar, but SSIM is not. The SSIM value of CP–FW is similar to CP, but CP–AFW is smaller than the other algorithms, which means that the CP–AFW algorithm for solving the minimizing energy functional for denoising may lose image structural information. In conclusion, CP has a high calculation cost and an excellent denoising performance, but CP–AFW has a low calculation cost and slightly poorer denoising performance than CP. CP–FW is a good algorithm to recommend. We take an image “Papav” from the dataset to show the performance of Gaussian denoising with different algorithms in Figure 7 and the image “Peppers” to show the performance of Poisson denoising in Figure 8. The experiments show that the new accelerated Chambolle algorithms have better trade-off between the computational cost and denoising.
Figure 7. Papav. (a) Original image; (b) Gaussian noise image; (c) CP for denoising; (d) CP–FW for denoising; and (e) CP–AFW for denoising.

Figure 8. Peppers. (a) Original image; (b) Poisson noise image; (c) CP for denoising; (d) CP–FW for denoising; and (e) CP–AFW for denoising.

6. Conclusions

Along with the work of the Chambolle projection algorithm for Gaussian denoising, we have proposed the accelerated Chambolle projection based on the Frank–Wolfe method. New algorithms have a lower computation cost and can maintain their denoising performance while reducing computational complexity in image processing. When extending the Chambolle projection algorithm to the Poisson noise condition, as a matter of course, the accelerated Chambolle projection algorithms are also extended. The convergence of algorithms is proved under the Gaussian noise or the Poisson noise condition. Theory and experiments have shown that the new algorithms demonstrate practicability on the task of image processing. It is worth noting that our proposed algorithms can extend to other variant total variation models and other image-processing tasks, which is something we will implement as soon as possible.
Author Contributions: Conceptualization, W.W. and X.F.; methodology, W.W. and X.F.; software, W.W.; validation, W.W.; formal analysis, W.W.; writing—original draft preparation, W.W.; writing—review and editing, W.W. and X.F.; visualization, W.W.; supervision, X.F.; and funding acquisition, W.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China grant number 61772389.

Data Availability Statement: Publicly available datasets were analyzed in this study. Our image dataset includes set 12 and publicly available images

Acknowledgments: The authors would like to thank the National Natural Science Foundation of China (Grants 61772389) for supporting this research work.

Conflicts of Interest: The authors declare no conflict of interest.

References


