ADMM algorithm

LongZhang *

July 7, 2025

1 Imaging

We know that imaging (F) by an optical imaging system equals the ideal image (X) convolved with the point spread function (K):

$$F = K * X + n \tag{1}$$

where n denotes random noise.

1.1 Why ADMM?

Why Direct Inverse Filtering Fails in Practical Non-Blind Image Restoration?

Theoretical Perfect Restoration: In theory, for a Linear Space-Invariant (LSI) system without noise, the perfect restoration formula is:

Fourier Transform $\rightarrow$ Divide by OTF $\rightarrow$ Inverse Fourier Transform

However, inverse filtering nearly always fails in practical non-blind image restoration due to:

1.2 OTF Zero Points:

1. The Optical Transfer Function (OTF, Fourier transform of PSF) contains frequency components where $OTF(\omega) = 0$.

2. Division by OTF at these points becomes infinite (1/0).

3. Even near-zero OTF values cause extreme amplification of corresponding frequency components.

1.3 Noise Amplification:

1. Real images always contain noise.

2. Noise predominantly resides in high-frequency regions.

3. OTF values are typically very small at high frequencies.

4. Division by small OTF values amplifies noise catastrophically.

1.4 PSF Truncation and Modeling Errors:

1. Measured PSFs are finite-sized, truncated approximations.

2. Truncation introduces artifacts in the OTF representation.

3. Accurate PSF modeling is challenging and error-prone.

1.5 Ill-Posed Nature:

1. Due to OTF zeros or near-zeros, restoration becomes ill-posed.

2. Small perturbations lead to massive variations in output.

2 Theory

2.1 Problem Formulation

Objective function:

$$\begin{aligned}\min_X \sum_i &\|D_i X\|_2 \\ &+ \frac{\mu}{2}\|K * X - F\|_F^2\end{aligned} \tag{2}$$

where $D_i X$ denotes the gradient of image X at pixel i, $\|\cdot\|_2$ is the vector 2-norm, and $\|\cdot\|_F$ is the Frobenius norm.

*Computational optical imaging; Picture processing

2.2 ADMM Formulation

Introduce auxiliary variables: Let $Y = (Y_1, Y_2)$ replace $DX = (D_x X, D_y X)$:

$$\min_{X,Y} \quad \|Y\|_{2,1} + \frac{\mu}{2}\|K * X - F\|_F^2 \tag{3}$$
$$\text{s.t.} \quad Y = DX \tag{4}$$

where $\|Y\|_{2,1} = \sum_{i,j}\sqrt{(Y_1)_{i,j}^2 + (Y_2)_{i,j}^2}$ is the isotropic TV norm.

Augmented Lagrangian: Introduce Lagrange multiplier $\Lambda = (\Lambda_1, \Lambda_2)$:

$$\begin{aligned}\mathcal{L}(X, Y, \Lambda) = \|Y\|_{2,1} &+ \frac{\mu}{2}\|K * X - F\|_F^2 \\ &+ \langle \Lambda, DX - Y\rangle \\ &+ \frac{\beta}{2}\|DX - Y\|_F^2\end{aligned} \tag{5}$$

where $\beta > 0$ is the penalty parameter.

2.3 ADMM Algorithm

(a) Update Y (Shrinkage step): Fix X and Λ, solve:

$$\min_Y \|Y\|_{2,1} + \frac{\beta}{2}\|DX - Y + \beta^{-1}\Lambda\|_F^2 \tag{6}$$

Let $Z = DX + \beta^{-1}\Lambda$. The problem becomes:

$$\min_Y \|Y\|_{2,1} + \frac{\beta}{2}\|Y - Z\|_F^2 \tag{7}$$

Derivation of the Shrinkage Solution: The optimization problem can be decomposed per pixel since both terms are separable. For each pixel (i, j), we solve:

$$\min_{v\in\mathbb{R}^2} \|v\|_2 + \frac{\beta}{2}\|v - z\|_2^2 \tag{8}$$

where $v = Y_{i,j}$, $z = Z_{i,j}$.

This is a proximal operator for the ℓ_2-norm. To solve it, we consider the subdifferential of the objective function $f(v) = \|v\|_2 + \frac{\beta}{2}\|v - z\|_2^2$:

- **Case 1:** $\|z\|_2 > \frac{1}{\beta}$

  The solution occurs when the gradient is zero:

  $$\frac{v}{\|v\|_2} + \beta(v - z) = 0$$
  $$v = \left(1 - \frac{1}{\beta\|z\|_2}\right) z$$

- **Case 2:** $\|z\|_2 \le \frac{1}{\beta}$

  The optimal solution is $v = 0$, as verified by the subdifferential condition:

  $$0 \in \partial f(0) = \{g - \beta z \mid \|g\|_2 \le 1\}$$
  $$\|\beta z\|_2 \le 1$$

Combining both cases, we obtain the vector shrinkage operator:

$$v = \frac{z}{\|z\|_2}\max\left(\|z\|_2 - \frac{1}{\beta}, 0\right) \tag{9}$$

Therefore, the solution for the entire image is:

$$Y_{i,j} = \frac{Z_{i,j}}{\|Z_{i,j}\|_2}\max\left(\|Z_{i,j}\|_2 - \frac{1}{\beta}, 0\right) \tag{10}$$

where $Z_{i,j} = ((Z_1)_{i,j}, (Z_2)_{i,j}) \in \mathbb{R}^2$.

(b) Update X (Linear subproblem): Fix Y and Λ, solve:

$$\min_X \frac{\mu}{2}\|K * X - F\|_F^2 + \frac{\beta}{2}\|DX - Y + \beta^{-1}\Lambda\|_F^2 \tag{11}$$

Optimality condition:

$$\mu K^*(K * X - F) + \beta D^*(DX - Y + \beta^{-1}\Lambda) = 0 \tag{12}$$

Rearranged form:

$$(\mu K^* K + \beta D^* D)X = \mu K^* F + \beta D^*(Y - \beta^{-1}\Lambda) \tag{13}$$

Frequency domain solution:

$$X = \mathcal{F}^{-1}\left(\frac{\mathcal{F}(\mu K^* F + \beta D^*(Y - \beta^{-1}\Lambda))}{\mu|\mathcal{F}(K)|^2 + \beta(|\mathcal{F}(D_x)|^2 + |\mathcal{F}(D_y)|^2)}\right) \tag{14}$$

Numerator implementation:

$$\text{Numerator} = \mu K^\top F - D^*(\Lambda) + \beta D^*(Y) \tag{15}$$

(c) Update multiplier:

$$\Lambda \leftarrow \Lambda - \gamma\beta(DX - Y) \tag{16}$$

where $\gamma \in (0, 1.618]$ is a relaxation parameter.

2.4 Precomputed Frequency Domain Constants

$\mathtt{H} = \mathcal{F}(H, \text{size}(F))$

$\mathtt{K*F} = \mathcal{F}^{-1}\left(\overline{\mathtt{H}} \cdot \mathcal{F}(F)\right)$

$|\mathcal{F}(D)|^2 = |\mathcal{F}([1, -1], \text{sizeF})|^2 + |\mathcal{F}([1; -1], \text{sizeF})|^2$

2.5 Finite Difference Operators

Forward difference:

$$(D_x U)_{i,j} = \begin{cases} U_{i,j+1} - U_{i,j} & j < n \\ U_{i,1} - U_{i,n} & j = n \end{cases}$$

$$(D_y U)_{i,j} = \begin{cases} U_{i+1,j} - U_{i,j} & i < m \\ U_{1,j} - U_{m,j} & i = m \end{cases}$$

Negative divergence ($D^* = -$div):

$$(D^*(V))_{i,j} = (V^x_{i,j} - V^x_{i,j-1}) + (V^y_{i,j} - V^y_{i-1,j}) \quad (17)$$

with circular boundary conditions:

$$V^x_{i,0} = V^x_{i,n}, \quad V^x_{i,n+1} = V^x_{i,1}$$
$$V^y_{0,j} = V^y_{m,j}, \quad V^y_{m+1,j} = V^y_{1,j}$$

2.6 Algorithm

Algorithm 1 ADMM for Image Restoration

Input: Blurred image F, kernel K, parameters μ, β, γ

Initialize: Restored image X

```
Initialize X^0 = F, Λ^0 = 0
for k = 0, 1, 2, ... do
    Update Y^{k+1} via shrinkage: Eq.(12)
    Update X^{k+1} via frequency domain: Eq.(16)

    if ‖X^{k+1} − X^k‖/‖X^k‖ < ε then
        break
    end if
    Update multiplier: Λ^{k+1} = Λ^k − γβ(DX^{k+1} − Y^{k+1})
end for
Output: Restored image X
```