A deep primal-dual proximal network for image restoration

Mingyuan Jiu and Nelly Pustelnik

School of Information Engineering, Zhengzhou University, Zhengzhou, 450001, China
CNRS, Laboratoire de Physique de l’ENS de Lyon, France

arXiv: 2007.00959
Motivation: image restoration

Direct model

\[ z = A\bar{x} + \varepsilon \]

where

- original image \( \bar{x} \in \mathbb{R}^N \) composed with \( N \) pixels
- \( A \in \mathbb{R}^{M \times N} \) models a linear degradation,
- \( \varepsilon \sim \mathcal{N}(0, \alpha^2 \mathbb{I}) \) models the effect of a white Gaussian noise of standard deviation \( \alpha \),
- \( z \in \mathbb{R}^M \) denotes the observed data.

Penalized likelihood approach

\[ \hat{x}_\lambda \in \text{Argmin}_x \frac{1}{2}\|Ax - z\|_2^2 + \lambda p(Hx), \]
Motivation: image restoration

Considered framework

\[ \hat{x}_\lambda \in \text{Argmin}_x \frac{1}{2} \|Ax - z\|_2^2 + g(Lx) \]

Parameter estimation

- Unsupervised techniques (i.e. SURE)

\[ \min_{\lambda} \mathbb{E}\{\|A(\hat{x}_\lambda - x)\|_2^2\}, \]

- Supervised techniques relies on a training data set

\[ S = \{(\bar{x}_s, z_s) | s = 1, \ldots, l}\]

\[ E(\Theta) := \frac{1}{l} \sum_{s=1}^{l} \|f_{\Theta}(\ell_{z}(z_s)) - \ell_x(\bar{x}_s)\|_2^2. \]
Primal-dual proximal algorithm

Considered framework

\[ \hat{x}_\lambda \in \text{Argmin}_x \frac{1}{2} \|Ax - z\|_2^2 + g(Lx) \]

Condat-Vũ primal-dual proximal algorithm

**Algorithm 1**: Primal-dual splitting algorithm for solving Problem 1.

1. **Set**: \( \tau > 0, \sigma > 0, \) such that \( \frac{1}{\tau} - \sigma \|L\|^2 \geq \frac{\|A\|^2}{2} \)
2. **Initialization**: \((x^{[1]}, y^{[1]}) \in \mathcal{H} \times \mathcal{G}\)
3. **for** \( k = 1, \ldots, K \) **do**
4. \[ x^{[k+1]} = x^{[k]} - \tau A^*(Ax^{[k]} - z) - \tau L^*y^{[k]} \]
5. \[ y^{[k+1]} = \text{prox}_{\sigma g^*} \left( y^{[k]} + \sigma L(2x^{[k+1]} - x^{[k]}) \right) \]
6. **end**
Proposition

Algorithm 1 fits the network

\[ u^K = \eta^K (D^K \ldots \eta^1 (D^1 u^1 + b^1) \ldots + b^K) \]

when considering, for every \( k \in \{1, \ldots, K\} \), \( D^k \in \mathbb{R}^{(N+P) \times (N+P)} \), \( b^k \in \mathbb{R}^{N+P} \) and \( \eta^k : \mathbb{R}^{N+P} \rightarrow \mathbb{R}^{N+P} \) such that

\[
\begin{align*}
D^k &= \begin{pmatrix}
Id - \tau A^* A & -\tau L^* \\
\sigma L(Id - 2\tau A^* A) & Id - 2\tau \sigma LL^*
\end{pmatrix} \\
b^k &= \begin{pmatrix}
\tau A^* z \\
2\tau \sigma LA^* z
\end{pmatrix} \\
\eta^k &= \begin{pmatrix}
Id \\
\text{prox}_{\sigma g^*}
\end{pmatrix}
\end{align*}
\] (1)

where \( \text{Id} \) denotes the identity matrix.
Proposed DeepPDNet

Proposition

Given the training set $S = \{(\bar{x}_s, z_s)|s = 1, \ldots, l\}$ where $\bar{x}_s$ is the undegraded image and $z_s = A\bar{x}_s + \varepsilon$ is its degraded counterpart. We build an inverse problem solver $f_\Theta$ relying on the estimation of $\hat{\Theta} = \{\hat{\sigma}^{[k]}, \hat{\tau}^{[k]}, \hat{L}^{[k]}\}_{1 \leq k \leq K}$:

$$\hat{\Theta} \in \text{Argmin}_{\Theta} \frac{1}{l} \sum_{s=1}^{l} \|\bar{x}_s - f_\Theta(z_s)\|_2^2 \quad \text{where} \quad f_\Theta(z_s) = \eta^{[K]}(D^{[K]} (\ldots \eta^{[1]}(D^{[1]} u^{[1]}_s + b^{[1]}) \ldots + b^{[K]}))$$

with

$$
\begin{align*}
    u^{[1]}_s &= A^* z_s \\
    D^{[1]} &= \begin{pmatrix}
    \text{Id} - \tau^{[1]} A^* A \\
    \sigma^{[1]} L^{[1]}(\text{Id} - 2\tau^{[1]} A^* A)
    \end{pmatrix} \\
    D^{[k]} &= \begin{pmatrix}
    \text{Id} - \tau^{[k]} A^* A & -\tau^{[k]} (L^{[k]})^* \\
    \sigma^{[k]} L^{[k]}(\text{Id} - 2\tau^{[k]} A^* A) & \text{Id} - 2\tau^{[k]} \sigma^{[k]} L^{[k]}(L^{[k]})^*
    \end{pmatrix} \\
    b^{[k]} &= \begin{pmatrix}
    \tau^{[k]} A^* z \\
    2\tau^{[k]} \sigma^{[k]} L^{[k]} A^* z
    \end{pmatrix} \\
    \eta^{[k]} &= \begin{pmatrix}
    \text{Id} \\
    \text{prox}_{\sigma^{[k]} g^*}
    \end{pmatrix} \\
    D^{[K]} &= \begin{pmatrix}
    \text{Id} - \tau^{[K]} A^* A & \tau^{[K]} (L^{[K]})^*
    \end{pmatrix} \\
    b^{[K]} &= \tau^{[K]} A^* z, \quad \eta^{[K]} = \text{Id}. 
\end{align*}
$$
Results

Figure 3. Visual comparisons on MNIST dataset for different methods. The first row corresponds to the MNIST data with a uniform $3 \times 3$ blur and a Gaussian noise with $\alpha = 20$, the second row is with a uniform $5 \times 5$ blur and a Gaussian noise with $\alpha = 20$, the third row is with a uniform $7 \times 7$ blur and a Gaussian noise with $\alpha = 20$. For each instance, the images from the first to the seventh column respectively correspond to the original one $\tilde{x}$, the degraded one $z$, the restored ones by EPLL, TV, NLTV, IRCNN and the proposed full DeepPDNet ($K = 6$).
## Results

| Data  | Method     | $3 \times 3$ Blur | $5 \times 5$ Blur | $7 \times 7$ Blur |
|-------|------------|-------------------|-------------------|-------------------|
|       |            | $\alpha = 10$    | $\alpha = 20$    | $\alpha = 30$    | $\alpha = 20$    |
|       |            | PSNR/SSIM         | PSNR/SSIM         | PSNR/SSIM         | PSNR/SSIM         |
| MNIST | EPL [18]   | 24.02/0.8564      | 20.99/0.7628      | 19.05/0.6871      | 16.42/0.5629      | 13.97/0.3265      |
|       | TV [8]     | 25.07/0.8583      | 19.58/0.7004      | 18.86/0.6681      | 18.86/0.6681      | 16.31/0.5665      |
|       | NLTV [53]  | 25.49/0.8697      | 21.98/0.7738      | 20.73/0.7353      | 20.73/0.7353      | 16.79/0.6228      |
|       | IRCNN [54] | **28.52/0.8904**  | 25.00/0.8193      | **22.63/0.7723**  | 21.46/0.7698      | 18.29/0.6546      |
|       | Partial DeepPDNet | 23.67/0.8366 | 22.03/0.7983 | 20.93/0.7750 | 17.96/0.6534 | 16.21/0.5505 |
|       | Full DeepPDNet | **27.40/0.9410** | **25.09/0.9254** | **23.61/0.9097** | **22.43/0.8738** | **20.43/0.8157** |

**Table III**

Comparison results of different methods on the MNIST dataset from different degradation configurations.
Main conclusions

- DeepPDNet allows us to achieve comparable performance than state-of-the-art CNN based strategies.

- We learn the algorithm step-size and $L$ (including the regularization parameter).

- If the network is designed to satisfy $\frac{1}{\tau} - \sigma \|L\|^2 \geq \frac{\|A\|^2}{2}$ (convergence guarantees of Condat-Vũ proximal scheme) the performance are significantly decreased.

- The more complex is the database, the more layer is required: MNIST 6 layers and BSD68 20 layers.