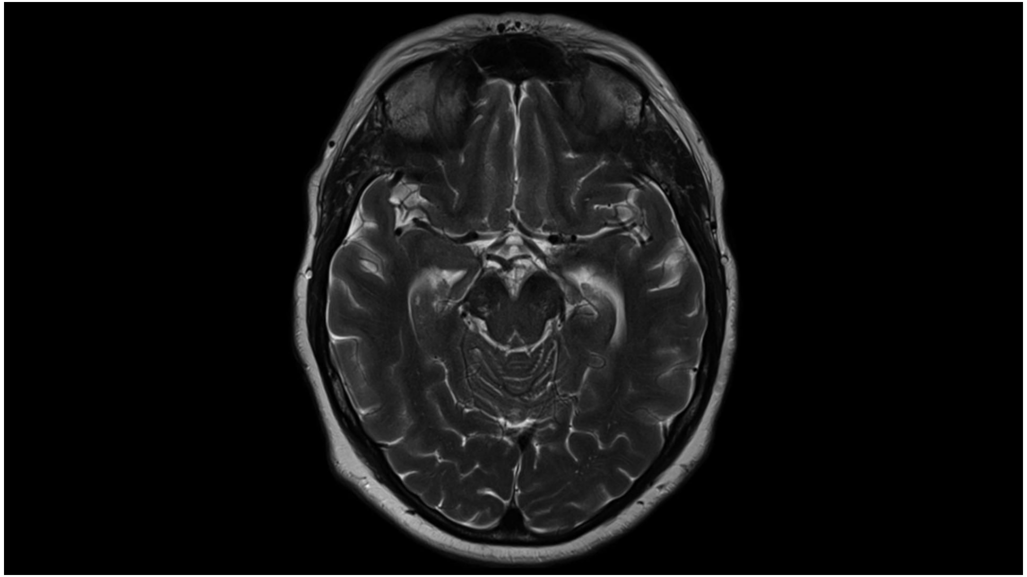
Total variation method for image reconstruction

Nikita Belousov
Dmitrii Masnyi

Skoltech, 2021

Why denoising is important?



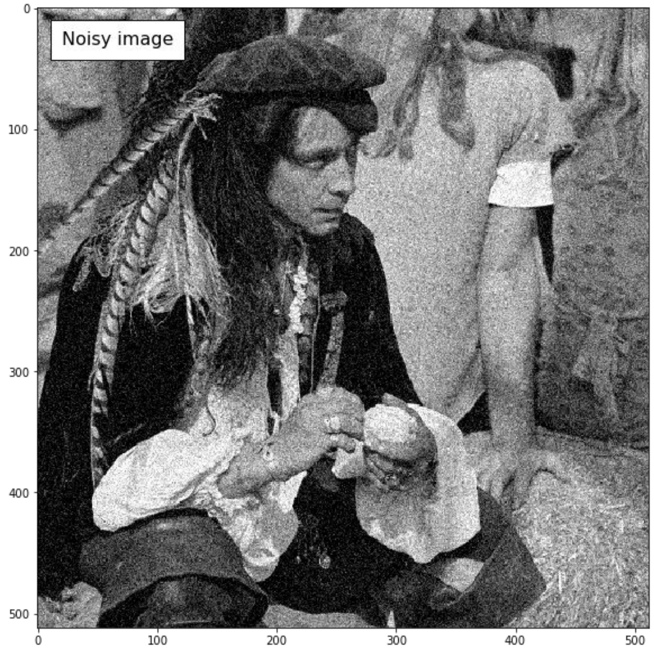
Classical methods

- Spatial domain filtering
- **Total variation regularization**
- Sparse representation
- Low-rank minimization

Notation

X

B = X + N



Total variation denoising problem

Primal problem:

$$\begin{aligned}&\text{minimize} \;\; \textstyle\sum_{i=1}^{m} \sum_{j=1}^{n} \| D_{(ij)}\, x \|_2 \\ &\text{subject to} \;\; \| x - b \|_2 \le \delta,\end{aligned}$$

Discrete TV function:

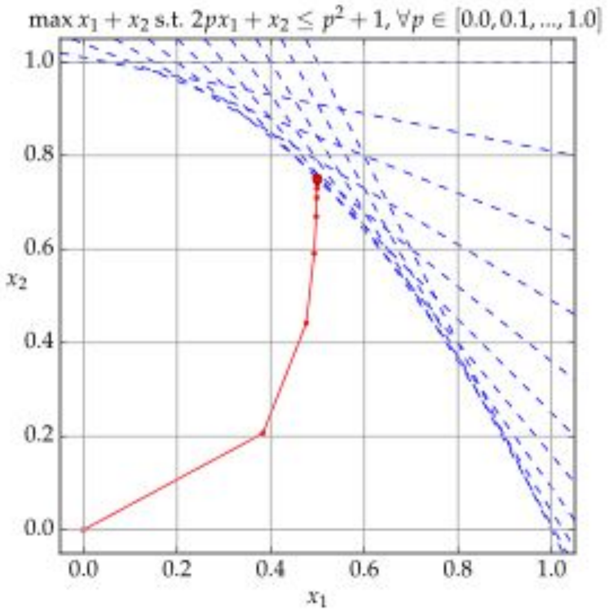
$$\mathcal{T}(x) = \sum_{i=1}^{m} \sum_{j=1}^{n} \| D_{(ij)}\, x \|_2,$$

Dual problem:

$$\begin{aligned}&\text{maximize} \;\; -\delta \, \| D^T u \|_2 + b^T D^T u \\ &\text{subject to} \;\; \| u_{(ij)} \|_2 \le 1, \quad i = 1, \ldots, m, \;\; j = 1, \ldots, n,\end{aligned}$$

Second-order cone programming problem (SOCP)

Classical method



Interior-point algorithm

Useless for large scale problems.

Nesterov’s first order algorithm

Fast approach

$$\min_{x \in Q_{\mathrm{p}}} \max_{u \in Q_{\mathrm{d}}} u^T D\, x,$$

$$Q_{\mathrm{p}} = \{x \mid \|x - b\|_2 \le \delta\},$$

$$Q_{\mathrm{d}} = \left\{u \mid \|u_{(ij)}\|_2 \le 1,\ i = 1, \ldots, m,\ j = 1, \ldots, n\right\}.$$

$$f_{\mathrm{p}}(x) = \tfrac{1}{2}\|x - b\|_2^2 \qquad \text{and} \qquad f_{\mathrm{d}}(u) = \tfrac{1}{2}\|u\|_2^2.$$

Fast approach

A smooth approximation

$$\mathcal{T}_\mu(x) = \max_{u \in Q_\mathrm{d}} \left\{ u^T D\, x - \mu\, f_\mathrm{d}(u) \right\}.$$

Terminating criteria:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} \| D_{(ij)}\, x \|_2 + \delta \| D^T u \|_2 - u^T D\, b < \epsilon.$$

Nesterov's first order algorithm

Given data b and a tolerance ϵ.

Set $x^{[0]} = b$ (a feasible starting point), $\mu = \frac{\epsilon}{2\Delta_{\mathrm{d}}}$, and $\mathcal{L}_\mu = \frac{\|D\|_2^2}{\mu}$.

For $k = 0, 1, 2, \ldots$

1) Evaluate $g^{[k]} = \nabla \mathcal{T}_\mu(x^{[k]})$.

2) Find $y^{[k]} = \arg\min_{x \in Q_{\mathrm{p}}} \left\{ (x - x^{[k]})^T g^{[k]} + \frac{1}{2}\mathcal{L}_\mu \|x - x^{[k]}\|_2^2 \right\}$.

3) Find $z^{[k]} = \arg\min_{x \in Q_{\mathrm{p}}} \left\{ \mathcal{L}_\mu \; f_{\mathrm{p}}(x) + \sum_{i=0}^{k} \frac{i+1}{2}(x - x^{[i]})^T g^{[k]} \right\}$.

4) Update $x^{[k+1]} = \frac{2}{k+3} z^{[k]} + \frac{k+1}{k+3} y^{[k]}$.

Modifications of the steps

Step 1

$$\nabla \mathcal{T}_\mu(x^{[k]}) = D^T u^{[k]},$$

$$u^{[k]} = \arg\max_{u \in Q_\mathrm{d}} u^T D\, x^{[k]} - \frac{\mu}{2} \|u\|_2^2.$$

$$u_{(ij)}^{[k]} = D_{(ij)} x^{[k]} / \max\left\{\mu,\, \|D_{(ij)} x^{[k]}\|_2\right\}.$$

Modifications of the steps

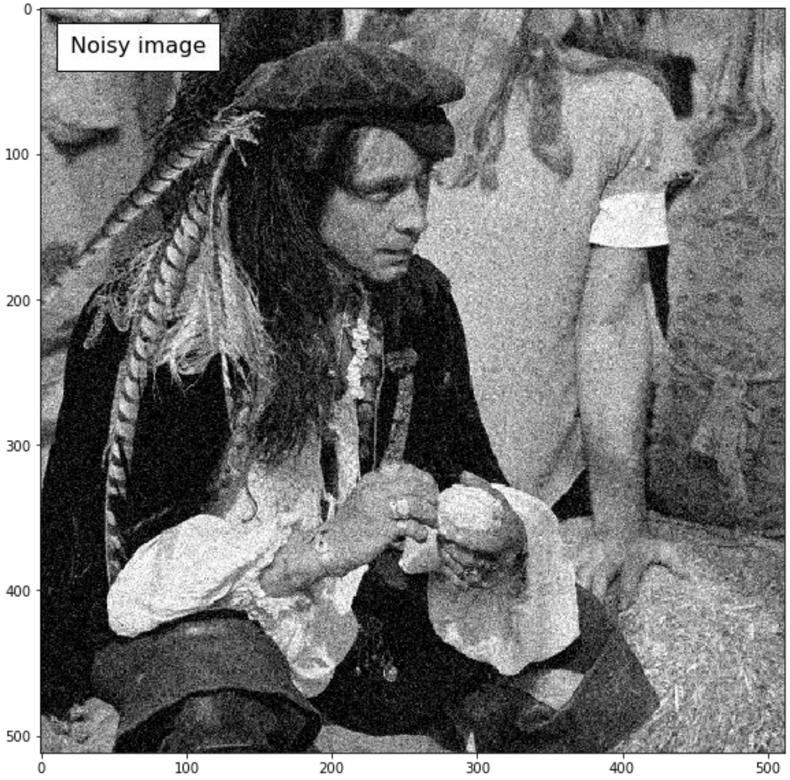
Step 2

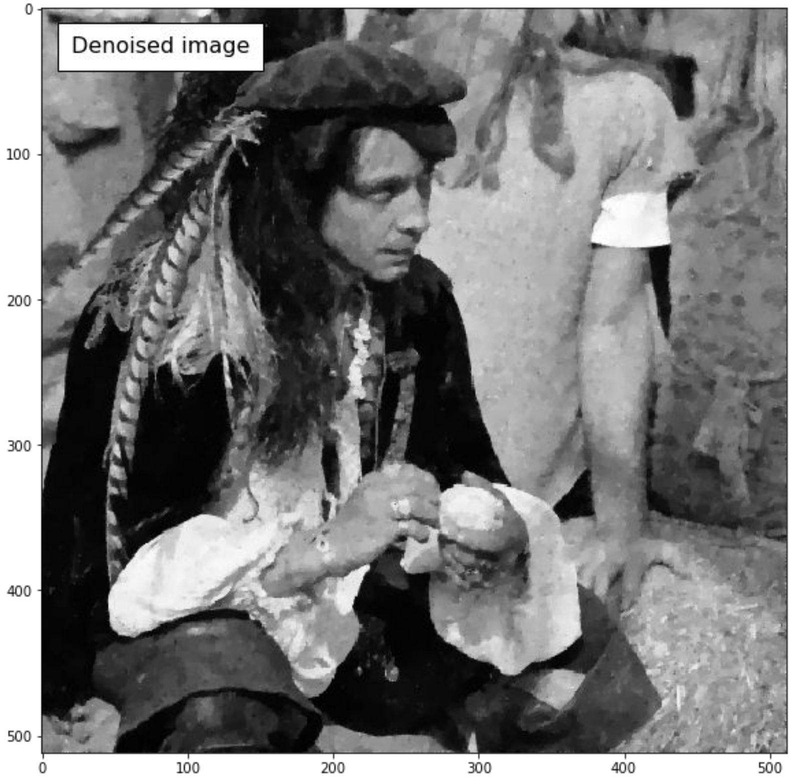
$$y^{[k]} = \left(\mathcal{L}_\mu \left(x^{[k]} - b\right) - g^{[k]}\right) / \max\left\{\mathcal{L}_\mu, \left\|\mathcal{L}_\mu \left(x^{[k]} - b\right) - g^{[k]}\right\|_2 / \delta\right\} + b,$$

Step 3

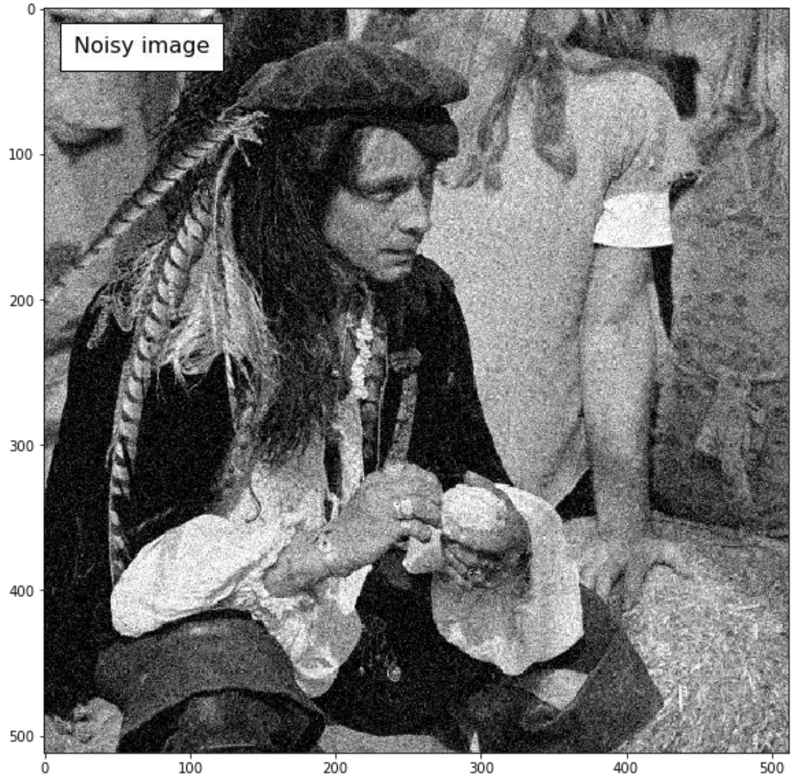
$$z^{[k]} = -w^{[k]} / \max\left\{\mathcal{L}_\mu, \left\|w^{[k]}\right\|_2 / \delta\right\} + b, \quad w^{[k]} = \sum_{i=0}^{k} \frac{1}{2}(i+1)\, g^{[i]}.$$

Denoising results: τ = 0.85, σ = 25

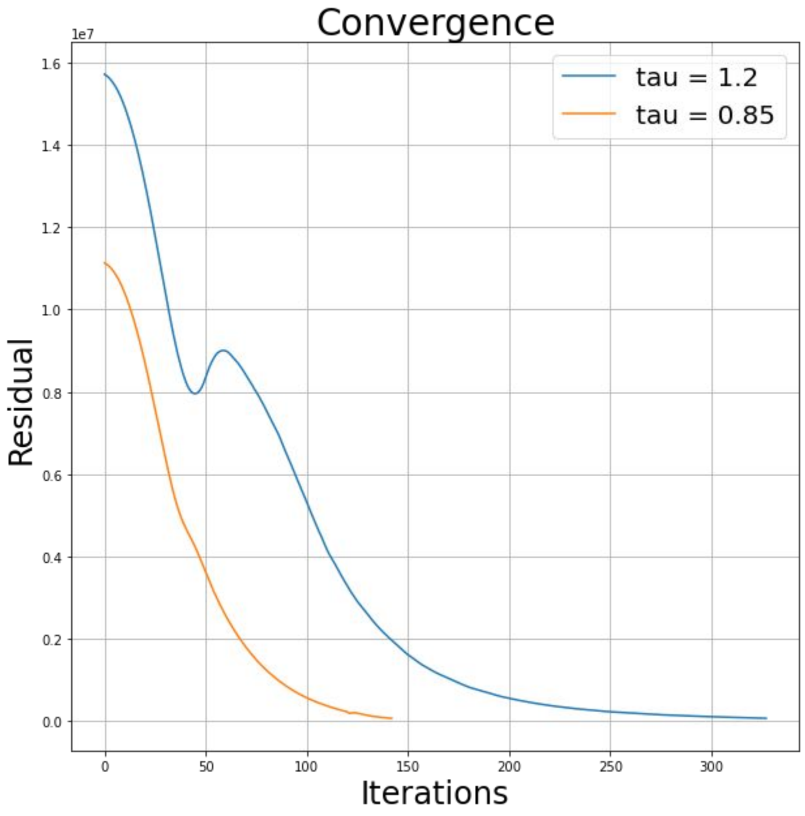


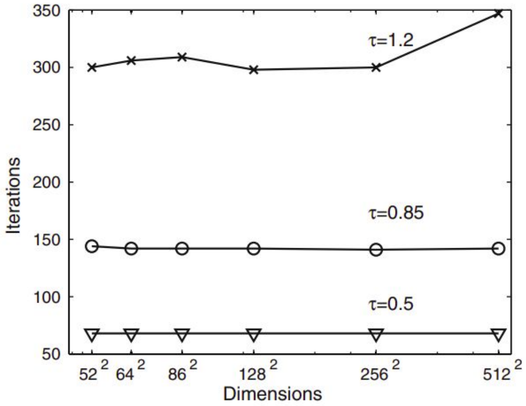


Denoising results: τ = 1.2, σ = 25



Simulation results





	τ = 0.85	τ = 1.2
N iterations	142	327
Time	4:12	9:56

Algorithm improvement

$$2)\ \text{Find } y^{[k]} = \arg\min_{x \in Q_{\mathrm{p}}} \left\{ (x - x^{[k]})^T g^{[k]} + \tfrac{1}{2} \mathcal{L}_\mu \|x - x^{[k]}\|_2^2 \right\}.$$

$$3)\ \text{Find } z^{[k]} = \arg\min_{x \in Q_{\mathrm{p}}} \left\{ \mathcal{L}_\mu\ f_{\mathrm{p}}(x) + \textstyle\sum_{i=0}^{k} \tfrac{i+1}{2} (x - x^{[i]})^T g^{[k]} \right\}.$$

	In parallel	Sequentially
Time	4:12	4:32

Conclusion

- Image denoising is a hot research topic. Over the years many techniques of image denoising have been introduced.
- We studied one of the first efficient algorithms for this problem.
- We write the code for TV denoising algorithm from the scratch, so it is easy for user to understand how this algorithm works under-the-hood.
- Parallelizing the calculation of step 2 and step 3 decreased the time for computation.
- https://github.com/dmasny99/cvx_opt_skoltech_project

Thanks for your attention!