Post-Composition Primal-Dual Algorithm for Texture, Geometry, and Noise Separation

Adrien Meynard[1], Nelly Pustelnik[1],
Luis Briceño-Arias[2] and Sylvain Meignen[3]

[1] *ENS de Lyon, CNRS, LPENSL, Lyon, France*

[2] *Universidad Técnica Federico Santa María, Departamento de Matemática, Santiago, Chile*

[3] *University Grenoble Alpes, CNRS, Jean Kuntzmann Laboratory, Grenoble, France*

1 Introduction

2 Problem Formulation

3 Algorithmic Solutions

4 Results

5 Conclusion and Perspectives

Motivation

Model-based image restoration

↓

Unfolded neural network architecture

- **Underlying structure of variational problem** is of considerable importance.
- Designing **fast algorithms** makes it possible to obtain efficient recovery.

Our Contribution

- Inverse problem = **Texture–Geometry–Noise decomposition**.
- Proposition:
  - Design a flexible variational formulation;
  - Construct a suitable and efficient algorithmic implementation.
- ⇒ **Building blocks for future unfolded framework.**

1 Introduction

2 Problem Formulation

3 Algorithmic Solutions

4 Results

5 Conclusion and Perspectives

Texture + Geometry + Noise model

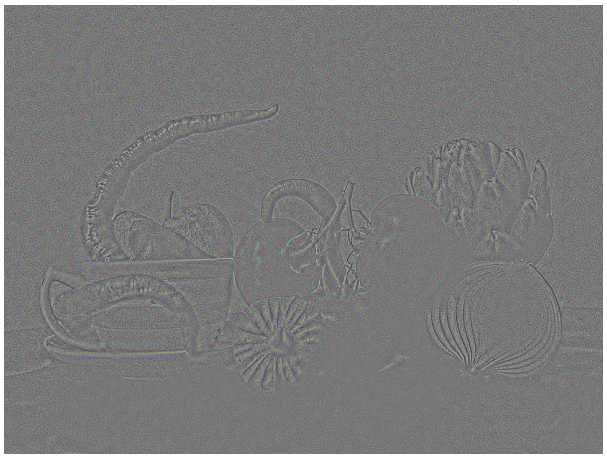
z: Original image



x: Geometry Component



y: Texture Component



Underlying assumption

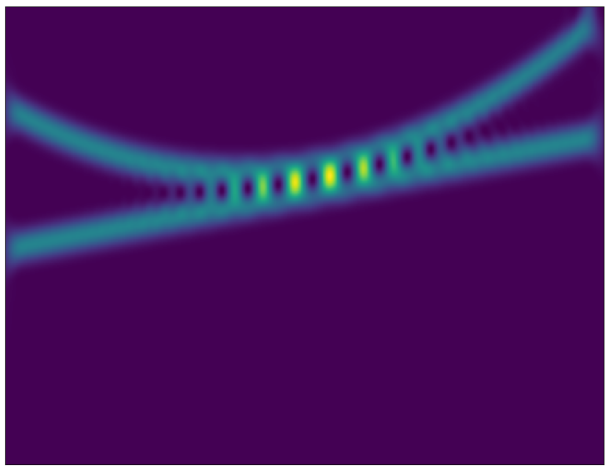
Additive Decomposition:

$$z = x + y + n,$$

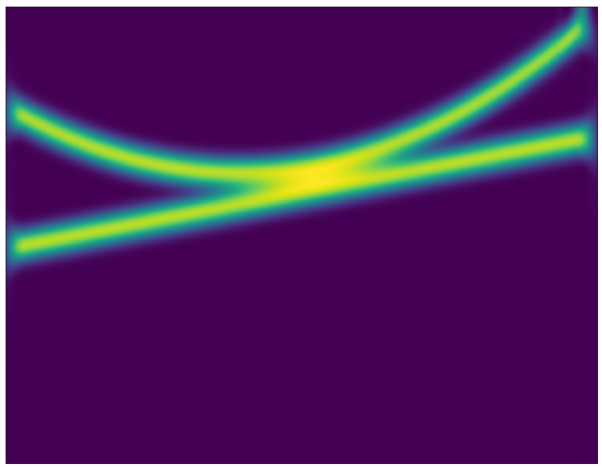
where n is an additive Gaussian white noise.

Texture + Geometry + Noise model

z: Original Spectrogram



x: Geometry Component



y: Texture Component



Underlying assumption

Additive Decomposition:

$$z = x + y + n,$$

Spectrograms are images of a particular nature: standard hypotheses about the components x, y and n may not be suited here.

Components Separation

Objective

Given the observation of the full noisy image z, separate the contributions of the geometry, texture, and noise components.

Proposed method = Resolve an **inverse problem**.

Requires **priors** on the behavior of each component.

- Geometry: Piecewise constant image.
- Texture: Locally oscillating image.
- Noise: Frequencially-spread spectrum.

Proposed Method: Inverse Problem

Resolve the following inverse problem:

$$\min_{(x,y,n)} \underbrace{\frac{1}{2}\|z-x-y-n\|_2^2}_{\text{Data-fidelity}} + \underbrace{\lambda J(x)}_{\substack{\text{Geometry}\\ \text{regularization}}} + \underbrace{J^*\left(\frac{y}{\mu}\right)}_{\substack{\text{Texture}\\ \text{regularization}}} + \underbrace{B^*\left(\frac{n}{\delta}\right)}_{\substack{\text{Noise}\\ \text{regularization}}}.$$

where

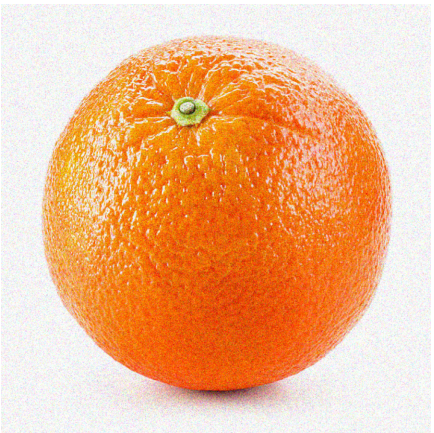
- J is the **image total variation** (TV);
- J^* is the **convex conjugate** of J;
- B^* is the **convex conjugate** of the ℓ^1-norm of wavelet coefficients.

Geometry Prior

Image total variation:

$$J(x) = \sum_{1 \leq i,j \leq N} \|(Lx)_{i,j}\|_2$$

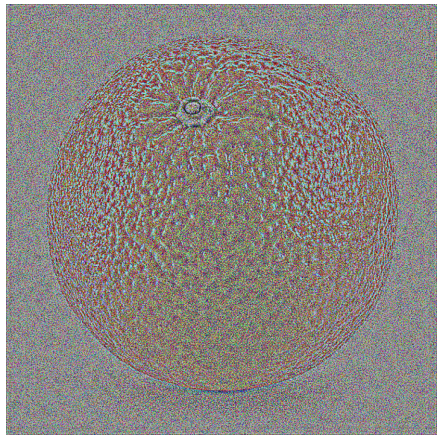
where L denotes the bi-dimensional finite differences operator.



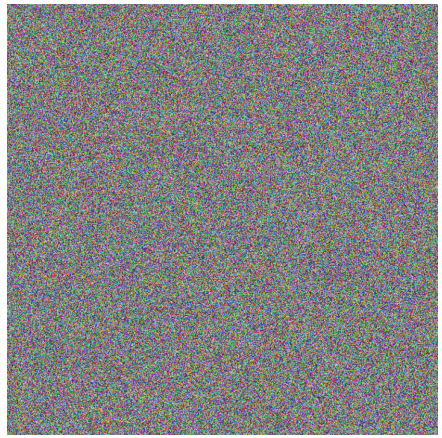
$J(z) = 1556$



$J(x) = 60$



$J(y) = 2190$



$J(n) = 828$

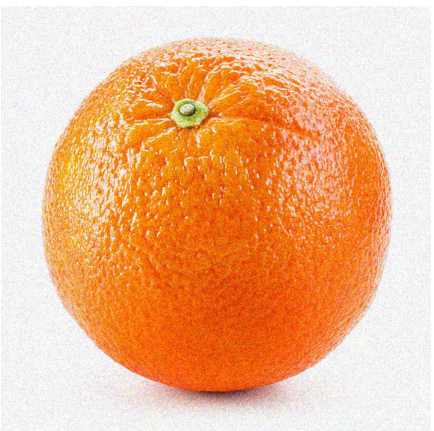
Texture Prior

J^* is the **indicator function** of the unit ball for the G norm, with

$$\|y\|_G = \inf \left\{ \|g\|_\infty = \max_{i,j} \|g_{i,j}\|_2, \; y = \operatorname{div}(g) \right\}.$$

Thus,

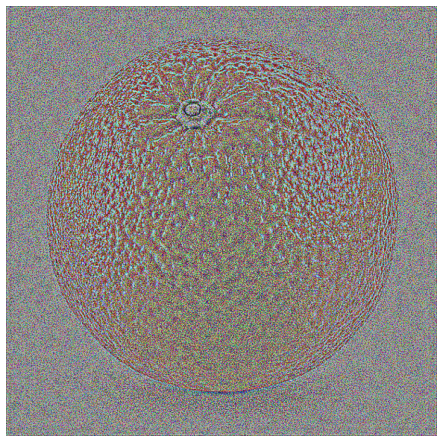
$$J^*(y) = \begin{cases} 0 & \text{if} \quad \|y\|_G \leq 1; \\ \infty & \text{if} \quad \|y\|_G > 1. \end{cases}$$



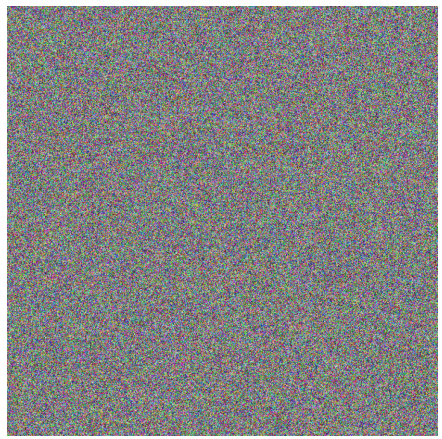
$\|z\|_G \sim 848.7$



$\|x\|_G \sim 848.3$



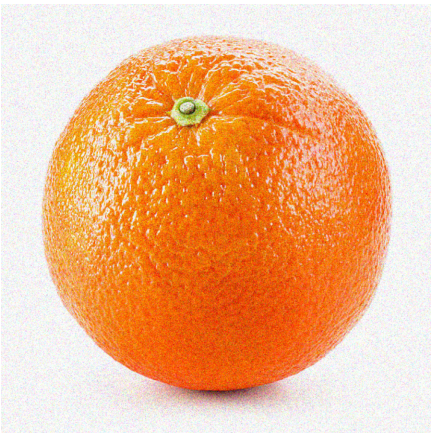
$\|y\|_G \sim 0.3$



$\|n\|_G \sim 0.1$

Noise Prior

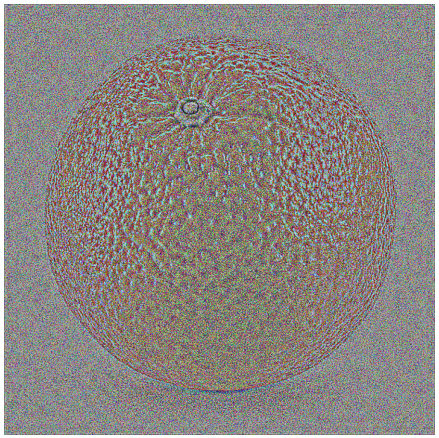
B^* is the **convex conjugate** of the ℓ^1-norm of wavelet coefficients.



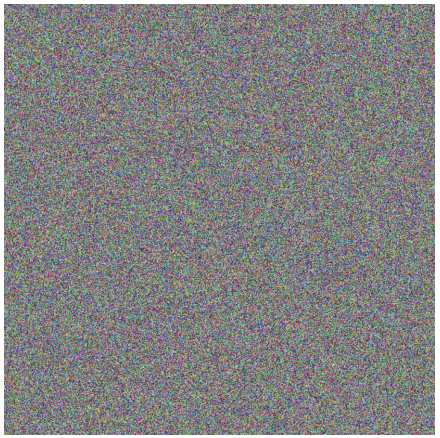
$\|z\|_{B^*} \sim 206$



$\|x\|_{B^*} \sim 42$



$\|y\|_{B^*} \sim 28$



$\|n\|_{B^*} \sim 9$

1 Introduction

2 Problem Formulation

3 Algorithmic Solutions

4 Results

5 Conclusion and Perspectives

Forward-Backward Algorithm

- Non-smooth involved functions $\Rightarrow$ **proximal schemes**.
- **Moreau decomposition** takes advantage of the dual formulation of the objective function:

$$\mathrm{prox}_{\gamma J^*(\cdot/\mu)}(x) = x - \frac{\gamma}{\mu}\mathrm{prox}_{\frac{\mu^2}{\gamma}J}\left(\mu\gamma^{-1}x\right).$$

Forward-Backward Algorithm

Updates for the kth iteration:

$$\begin{aligned} v_k &= x_k + y_k + n_k - z \\ x_{k+1} &= \mathrm{prox}_{\gamma\lambda J}(x_k - \gamma v_k) \\ y_{k+1} &= \mathrm{prox}_{\gamma J^*(\cdot/\mu)}(y_k - \gamma v_k) \\ n_{k+1} &= \mathrm{prox}_{\gamma B^*(\cdot/\delta)}(n_k - \gamma v_k). \end{aligned}$$

Major shortcoming: No closed-form expression for prox_J. Need inner iterations (FISTA) to approximate $\Rightarrow$ ↗ **Computing time!**

Post-Composition Primal-Dual Algorithm (1/2)

Because $J = f \circ L$ where $f = \|\cdot\|_{1,2}$, we can reformulate the inverse problem using the infimal **post-composition** operation, denoted by $\triangleright$:

$$(f \circ L)^* = L^* \triangleright f^* : t \mapsto \inf_{u,\, L^*u = t} f^*(u).$$

The minimization problem rewrites

$$\min_{x,u,n} \frac{1}{2}\|z - x - L^*u - n\|_2^2 + \lambda(f \circ L)(x) + f^*\left(\frac{u}{\mu}\right) + B^*\left(\frac{n}{\delta}\right)$$

with $L^*u = y$.

Post-Composition Primal-Dual Algorithm (2/2)

The primal-dual algorithm is derived from the **Condat-Vũ iterations**.

Post-Composition Primal-Dual Algorithm

- Updates for the kth iteration:

$$
\begin{aligned}
w_k &= x_k + y_k + n_k - z \\
x_{k+1} &= x_k - \tau\left(w_k + L^* v_k\right) \\
u_{k+1} &= \operatorname{prox}_{\tau f^*(\cdot/\mu)}\left(u_k - \tau L w_k\right) \text{ and } y_{k+1} = L^* u_{k+1} \\
n_{k+1} &= \operatorname{prox}_{\tau B^*(\cdot/\delta)}\left(n_k - \tau w_k\right) \\
v_{k+1} &= \operatorname{prox}_{\sigma(\lambda f)^*}\left(v_k + \sigma L(2x_{k+1} - x_k)\right).
\end{aligned}
$$

- To **ensure convergence**, σ and τ must satisfy

$$
\tau < \frac{2}{2 + (1 + 2\sigma)\|L\|^2}.
$$

Primal-dual iterations now only involve the proximal operator of f (or f^*), which is known.

1 Introduction

2 Problem Formulation

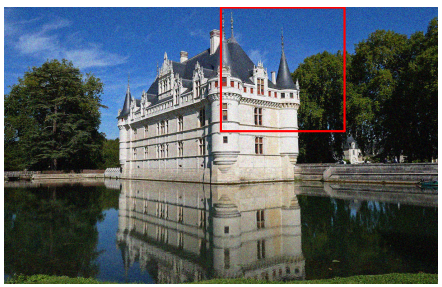
3 Algorithmic Solutions

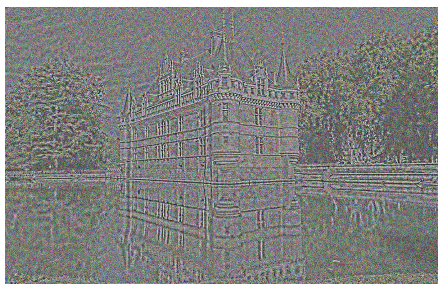
4 Results

5 Conclusion and Perspectives

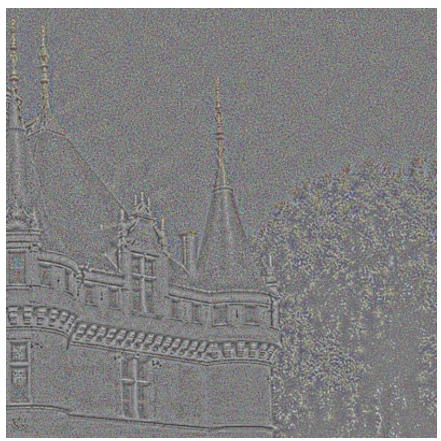
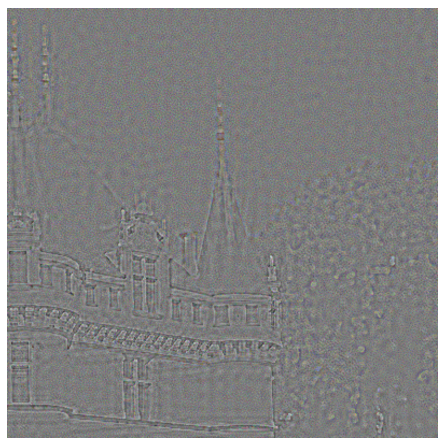
Qualitative Evaluation

- Implementation of the algorithms on a real 894×1400 image.
- Regularization parameters: $\lambda = 0.01$, $\mu = 0.2$ and $\delta = 0.01$, the value for the latter parameter being automatically computed.





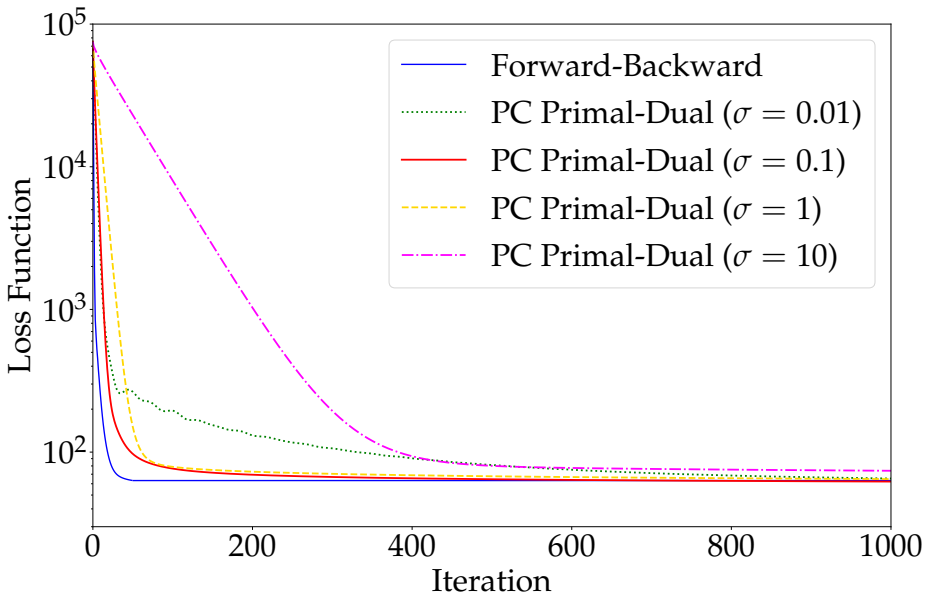


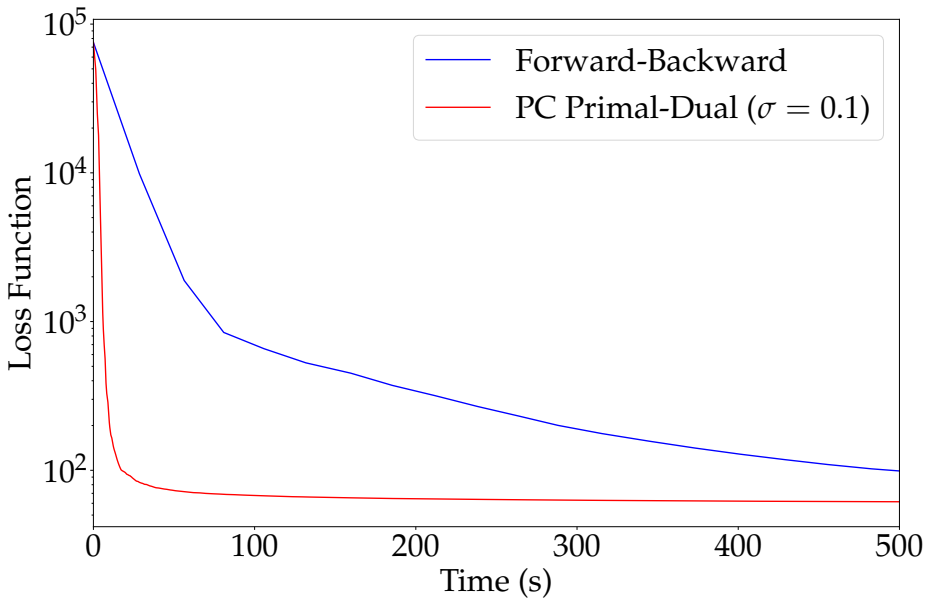
(a) Original (b) Geometry (c) Texture (d) Noise

Computational Time Evaluation

- Comparison of the speed of convergence of the minimization algorithms.
- Performed on the zoom-in view as the computation with FB was too long to process for the full image.

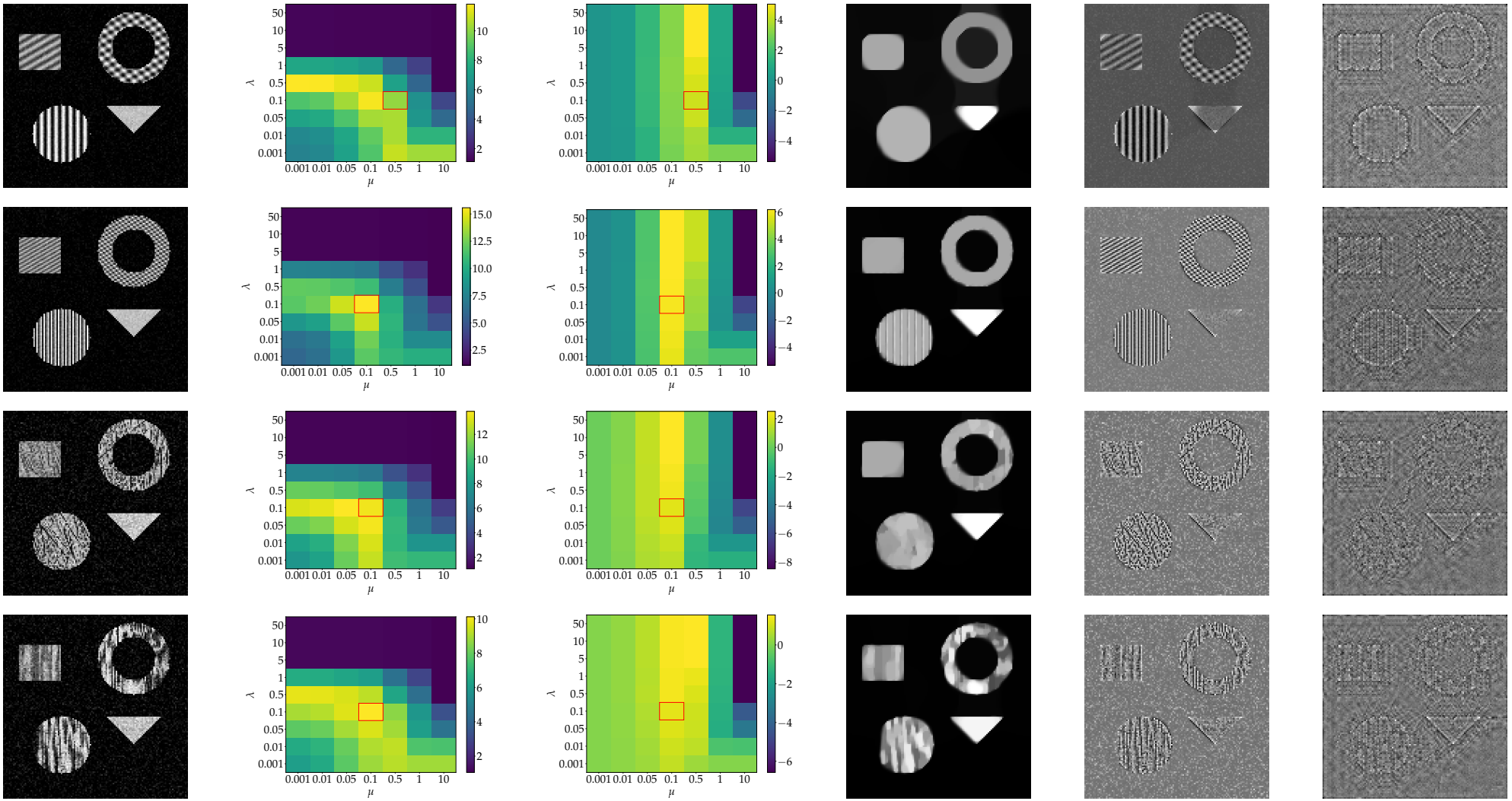


(a) Iteration convergence



(b) Computation time

Role of the Regularization Parameters and Limitations



(a) Original (b) Geometry SNR (c) Texture SNR (d) Geometry (e) Texture (f) Noise

1 Introduction

2 Problem Formulation

3 Algorithmic Solutions

4 Results

5 Conclusion and Perspectives

Conclusion and Perspectives

- New and **faster algorithmic scheme** for Texture–Geometry–Noise decomposition, leveraging post-composition primal-dual iterations with **convergence guarantees**.
- Paves the way to
  - extended analysis on the choice of L and f;
  - the design of unfolded schemes in order to learn more suitable norms for texture and geometry decomposition.