Optimization in Image Restoration: Denoising and Deblurring Methods

Gonzalez Dardik, M. Ryszfeld, S.

Telecom Paris – Introduction to Image Processing

Abstract

This report presents an initial exploration into image restoration, focusing on how various parameters and regularization methods influence restoration quality. The primary objective is to implement optimization techniques for image restauration, using basic methods, such as Tykhonov's regularization (implemented both analytically in the Fourier domain and numerically via gradient descent in PyTorch) and Total Variation (TV) regularization.

The first part of the report addresses only the **denoising** part of the restauration (which assumes that the image does not have blur), and the second part consists of the **deblurring** processes (implying the image has both blur and noise)

Evaluation of restoration performance combines qualitative visual inspection with quantitative metrics such as Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM).

1 Introduction

Image restoration seeks to recover a clean version of an image from a degraded observation. This task is commonly formulated as an inverse problem, where the goal is to reconstruct an unknown clean image by minimizing an objective function that balances fidelity to the observed data with prior assumptions about the image structure. Typically, this is achieved through a data fidelity term, which enforces consistency with the observed (degraded) image, and a regularization term, which stabilizes the solution by imposing smoothness or structural constraints.

In this work, we investigate two classical regularization approaches. **Tykhonov** regularization imposes a quadratic smoothness prior, favouring globally smooth solutions, while **Total Variation** (TV) regularization promotes sparsity of image gradients, making it better suited for preserving sharp edges and discontinuities. Tykhonov method is implemented using closed-form solutions in the Fourier domain as well as iterative optimization via gradient descent in PyTorch. The Fourier-domain solutions serve as a reference to validate the numerical implementations.

The study is organized progressively. We first consider the image **denoising** problem, where the degradation is comparatively simple, facilitating the analysis of algorithmic behaviour and parameter selection. We then extend the framework to the more challenging image **deblurring** problem, where the degradation operator is non-trivial and the inverse problem becomes significantly ill-posed.

1.1 Problem Setup

We adopt the standard linear image degradation model:

$$v = Au + w,$$

where u denotes the unknown clean image, A is the degradation operator, and $w \sim \mathcal{N}(0, \sigma^2)$ is Gaussian noise.

The restoration problem is formulated as a regularized inverse problem, seeking

$$\hat{u} = \arg\min_u \frac{1}{2}\|Au - v\|^2 + \lambda\,\mathcal{R}(u),$$

where $\lambda > 0$ is the regularization parameter controlling the trade-off between data fidelity and prior assumptions and $\mathcal{R}(u)$ denotes the regularization functional :

- Tykhonov: $\|\nabla u\|^2$
- Total Variation: $\sum_{i,j} \sqrt{(\nabla_x u_{i,j})^2 + (\nabla_y u_{i,j})^2 + \varepsilon^2}$

Two degradation scenarios are considered:

- **Denoising:** $A = I$, so the model reduces to $v = u + w$. A relatively high noise level ($\sigma = 0.5$) is chosen to illustrate the effect of regularization on reconstruction quality.
- **Deblurring:** A is a convolution operator with a known blur kernel, i.e., $v = A * u + w$. Here, a lower noise level ($\sigma = 0.02$) is used, since the blur dominates the degradation and higher noise could destabilize the inversion.

1.2 Quantitative Evaluation Metrics

The quality of the restored images is evaluated using several standard quantitative metrics that assess different aspects of reconstruction error and perceptual similarity.

The **Mean Squared Error (MSE)** and **Mean Absolute Error (MAE)** measure the pixel-wise difference between the original and reconstructed images, with MSE emphasizing larger errors and MAE treating all deviations equally.

The **Peak Signal-to-Noise Ratio (PSNR)** expresses the ratio between the maximum possible image intensity and the reconstruction error, providing a scale in decibels that indicates overall image fidelity.

The **Structural Similarity Index (SSIM)** evaluates perceptual similarity by comparing local patterns of luminance, contrast, and structure between the original and restored images, capturing aspects of image quality that are more aligned with human perception.

All metrics were computed using standard library implementations.

2 Denoising

Denoising is the process of removing noise from an image while preserving important features such as edges and textures. We focus on Tykhonov regularization—both analytically (Fourier domain) and numerically (gradient descent using PyTorch)—as well as Total Variation (TV) regularization.

2.1 Method 1: Tykhonov Regularisation

Tykhonov regularization aims to minimize the functional

$$J(u) = \frac{1}{2}\|u - v\|^2 + \lambda \|\nabla u\|^2,$$

where v is the noisy image and λ controls the smoothness of the reconstruction. The first term ensures fidelity to the observed data, while the second penalizes large gradients, enforcing spatial smoothness.

2.1.1 Analytical Tykhonov — Fourier Solution

In the Fourier domain, the solution can be expressed in closed form, which allows immediate computing of the restored image.

The choice of λ strongly influences the reconstruction: small values leave too much noise, while large values oversmooth the image and remove meaningful structures. A suitable value must balance noise removal and detail preservation.

Choice of the Regularization Parameter (λ) [1]

Qualitative Analysis. Visual inspection of denoised images for a range of λ values showed that values between $\lambda = 1$ and $\lambda = 3$ provided reconstructions smoother than the noisy input while retaining important structures. $\lambda \approx 2$ appeared to offer a good balance.

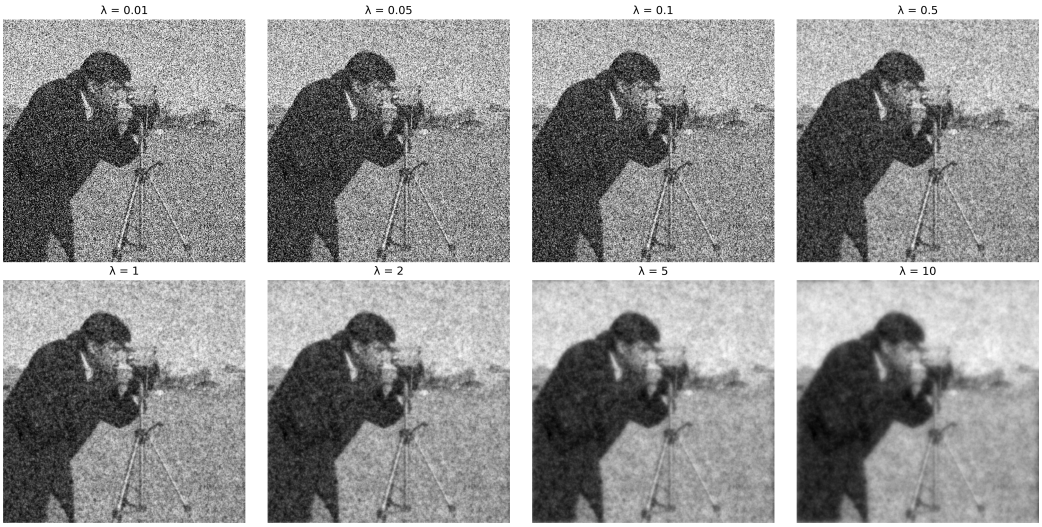


Figure 1: Tykhonov Fourier-domain denoising results for different values of λ.

Quantitative Analysis. We measured PSNR, SSIM, RMSE, and MAE for each value of λ. Although PSNR variations above $\lambda = 1$ remained relatively small, the metrics indicated that $\lambda = 2$ achieves the best compromise.

λ	PSNR [dB] (↑)	SSIM (↑)	RMSE (↓)	MAE (↓)
0.01	9.78	0.052	0.324	0.256
0.05	11.50	0.069	0.266	0.206
0.1	12.89	0.086	0.227	0.174
0.5	16.52	0.174	0.149	0.121
1	17.44	0.244	0.134	0.112
2	17.84	0.337	0.128	0.109
5	17.81	0.455	0.129	0.110
10	17.57	0.504	0.132	0.113

Table 1: Quantitative evaluation for several values of λ. Highlighted: $\lambda = 2$.

As part of the parameter selection process, A range of candidate values of the regularization parameter λ, was explored to identify values that produce stable and visually consistent results.

2.1.2 Numerical Tykhonov — Gradient Descent (PyTorch)

We minimized the Tykhonov functional numerically using gradient descent:

$$u^{(k+1)} = u^{(k)} - \eta \nabla J(u^{(k)}),$$

and compared it with the analytical Fourier solution to validate the implementation.

Figure 2 shows the evolution of the cost function. The loss decreases monotonically until convergence, confirming that the chosen learning rate is consistent with the theoretically optimal step.

[1] In this experiment we use a fixed noise level, therefore this configuration is adequate for this specific configuration.

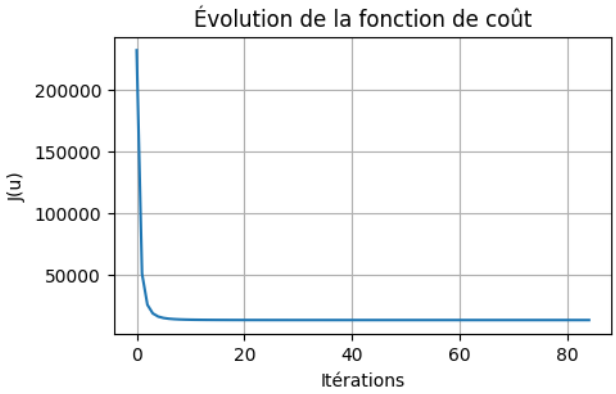


Figure 2: Evolution of the cost function during gradient descent optimization.

Qualitative Comparison - analytical and iterative approach. The reconstructions are visually indistinguishable (Figure 3), confirmed by image statistics in Table 2.

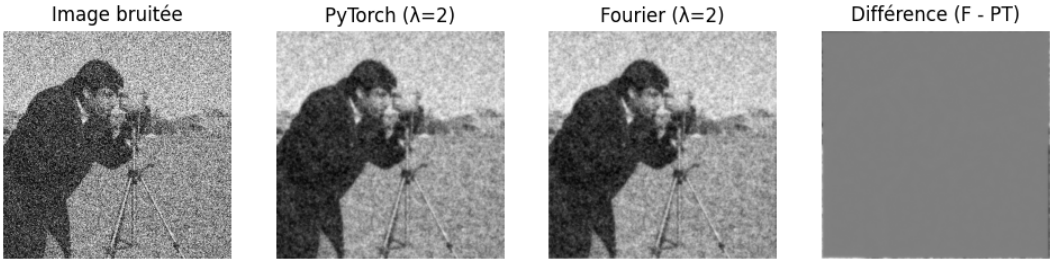
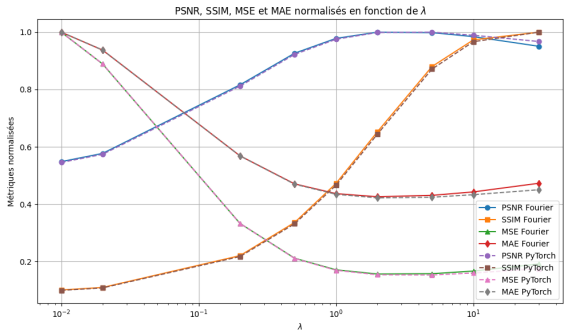


Figure 3: Comparison between Fourier and PyTorch Tykhonov results ($\lambda = 2$).

Image	Min	Max	Mean
Noisy	0.0000	1.0000	0.5071
PyTorch	0.1133	0.8519	0.5071
Fourier	0.1133	0.8519	0.5071

Table 2: Statistics for noisy input and reconstructions.

Quantitative Comparison - analytical and iterative approach. The final reconstructions using PSNR, SSIM, MSE and MAE are nearly identical for Fourier and PyTorch solutions, confirming convergence to the same solution.



Normalized metrics vs λ

Figure 4: Comparison of the normalized metrics as a function of the regularization parameter λ.

Observation. Subtracting the Fourier and PyTorch reconstructions produces a nearly uniform grey image, confirming that both solutions coincide. Minor non-uniformities appear near the borders due to periodic boundary assumptions in Fourier.

Choice of parameters. The tolerance parameter ε was initially set to 0.001, but was later increased to 0.01 after observing that this value provided sufficiently good results for the span of this study.

The learning rate was determined based on an estimate of the Lipschitz constant of the gradient:[2]

$$\text{lr_tych} = \frac{1}{1+16\lambda}.$$

Conclusion. Comparing both approaches show that they are equivalent which confirms that using the Pytorch implementation is adequate for this study.

2.2 Method 2: Total Variation Regularisation

The Total Variation (TV) regularization aims to minimize the functional:

$$J_{\text{TV}}(u) = \frac{1}{2}\|u-v\|^2 + \lambda \sum_{i,j} \sqrt{(\nabla_x u_{i,j})^2 + (\nabla_y u_{i,j})^2 + \epsilon^2},$$

where ϵ ensures differentiability.

2.2.1 Total Variation - Gradient Descent (PyTorch) solution

Gradient descent was used to minimize J_{TV}, with parameters chosen using theoretical derivation and experimental validation.

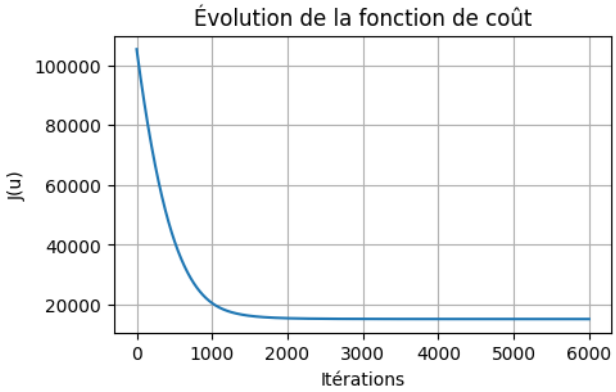


Figure 5: Evolution of the cost function during TV gradient descent optimization.

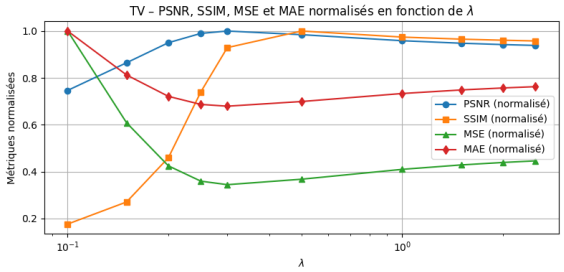
Computational cost. TV requires significantly more iterations than Tykhonov.

Choice of the Regularization Parameter λ The same methodology as in Tykhonov's process was used to find λ, this means performing qualitative experiments and quantitative measurements. Based on that, it was determined that an optimal λ value could be between 0.5 and 1. In this case, $\lambda = 0.7.$ was selected.

[2] A. Leclair, *Méthodes variationnelles pour la Restauration d'Images*, Telecom Paris, 2025.

Learning Rate. [3]:

$$\eta_{\mathrm{TV}} = \frac{2}{1 + 8\lambda/\epsilon}, \quad \epsilon = 10^{-3}.$$



Metrics vs η_{TV}

Figure 6: Metrics for TV denoising as a function of λ (via η_{TV}).

Optimal λ values were slightly lower in TV, than in Tykhonov, typically $\lambda \approx 0.5$–1.

Qualitative Differences between both regularisations. TV preserves better edges and sharp structures, while Tykhonov tends to blur fine details. In very smooth regions, TV may leave small residual noise patches.

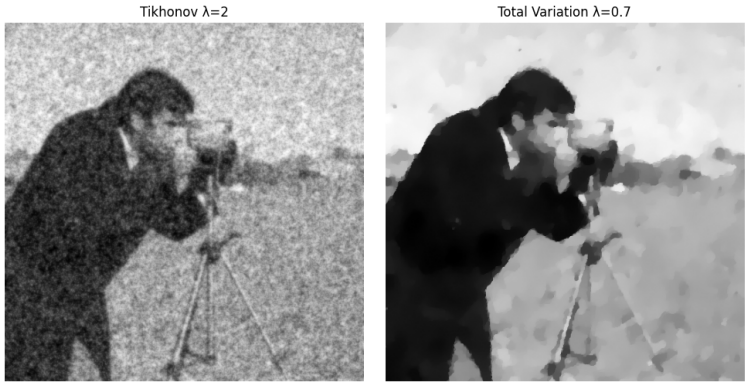


Figure 7: Visual comparison between Tykhonov and Total Variation outputs.

Quantitative Differences. PSNR isn't significantly higher than TV, but SSIM is typically much higher.

Method	λ	**PSNR [dB]**	**MSE**	**MAE**	**SSIM**
Tykhonov	2	17.90	0.0162	0.1084	0.335
Total Variation	0.7 ($\epsilon = 0.001$)	18.17	0.0153	0.1087	0.562

Table 3: Comparison of PSNR, MSE, MAE, and SSIM for Tykhonov and TV.

Limitations. Tykhonov oversmooths edges and cannot preserve sharp transitions, whereas TV may leave minor texture artifacts and requires careful tuning of ϵ and η. On balance, TV provides superior structural preservation but is more sensitive to hyperparameters.

[3] A. Leclair, *Méthodes variationnelles pour la Restauration d'Images*, Telecom Paris, 2025.

3 Deblurring

Deblurring aims to recover a clean image from an observation degraded by a known blur and additive noise. The problem is intrinsically ill-posed due to the non-invertibility of the blur operator and noise amplification in the frequency domain.

$$v = A * u + w,$$

3.1 Problem Formulation

We solve the variational problem

$$\min_{u \in [0,1]} \frac{1}{2}\|A * u - v\|^2 + \lambda \mathcal{R}(u),$$

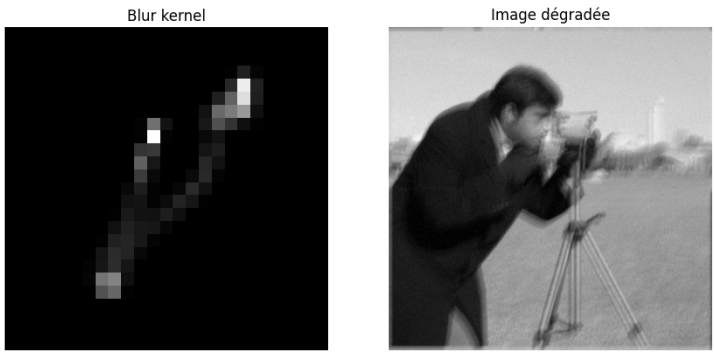
with the same choices of $\mathcal{R}$ used in denoising:

$$\mathcal{R}(u) = \|\nabla u\|^2 \quad \text{(Tykhonov)}, \qquad \mathcal{R}(u) = \sum_{i,j} \sqrt{|\nabla u_{i,j}|^2 + \varepsilon^2} \quad \text{(TV)}.$$

Since convolution is implemented via FFT in both the forward operator $k * u$ and its gradient $\nabla(k * u)$, the optimisation procedure is identical to the denoising case once the operator is defined.

3.2 Degradation Model

The degraded observation is generated by applying a known blur kernel to the ground truth image, followed by the addition of Gaussian noise. The blur kernel is first normalised such that its sum equals one and then shifted to ensure proper alignment in the frequency domain. Convolution is implemented via FFT, which efficiently computes the periodic convolution required for this setup. Finally, additive Gaussian noise with standard deviation $\sigma = 0.02$ is added to obtain the final degraded image.



Blur kernel Degraded image

3.3 Optimisation Setup

The reconstruction problem is solved using first-order gradient-based optimisation, analogous to the denoising case. Learning rates are derived analytically from the spectral properties of the forward and regularisation operators[4]:

$$\eta_{\text{Tikh}} = \frac{1}{\|A^\top A\| + 16\lambda}, \qquad \eta_{\text{TV}} = \frac{2}{\|A^\top A\| + 8\lambda/\varepsilon}.$$

Tykhonov regularisation converges rapidly, typically within tens of iterations, due to the convexity and smoothness of the associated functional. TV regularisation, being non-quadratic and edge-preserving, requires substantially more iterations to reach convergence.

[4] A. Leclair, *Méthodes variationnelles pour la Restauration d'Images*, Telecom Paris, 2025.

As in the denoising experiments, multiple configurations of λ, ε, and related parameters were explored; only the best-performing settings are reported here. The multiple tests can be found in the code

3.4 Parameters choice

As in the denoising part, we follow a two–step methodology combining a *qualitative inspection* and a *quantitative evaluation* of the reconstruction quality for several values of λ.

For each method (Tykhonov and Total Variation), we sweep a range of regularisation parameters and compare the resulting images visually and using standard metrics (PSNR, MSE, MAE, SSIM). This allows us to select the optimal parameters for each regulariser.

From the parameter study (see notes), the following values are chosen:

- **Tykhonov:** $\lambda = 0.5$
- **Total Variation:** $\lambda = 0.03$, $\varepsilon = 0.005$

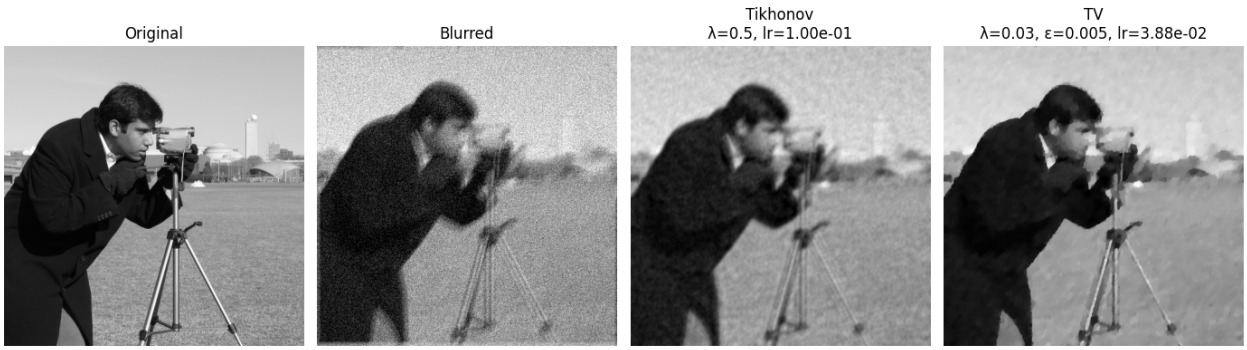


Figure 9: Deblurring results using Tykhonov and smoothed Total Variation for the selected parameters.

3.5 Loss Evolution

Both methods exhibit smooth, monotonically decreasing objective values, indicating stable optimisation. Tykhonov converges faster thanks to its quadratic regulariser, whereas TV shows a slower decrease due to the nonlinearity of the total variation term.

3.6 Qualitative assessment

Tykhonov regularisation produces smooth reconstructions where noise is effectively reduced but edges and fine structures are noticeably blurred. In contrast, TV regularisation preserves sharper contours and restores more geometric detail, yielding a reconstruction closer to the original image. Some mild artefacts can appear in flatter regions, but overall TV produces sharper reconstructions and better preserves the underlying image structure compared to Tykhonov.

3.7 Quantitative Evaluation

Table 4 summarises the reconstruction scores for the two regularisers using the optimal hyperparameters.

Method	λ	**PSNR [dB]**	**MSE**	**MAE**	**SSIM**
Tykhonov	0.5	23.52	0.0044	0.0421	0.578
Total Variation	0.03 ($\varepsilon = 0.005$)	25.09	0.0031	0.0337	0.668

Table 4: Quantitative comparison of deblurring performance for the selected regularisation parameters.

These results confirm the qualitative observations: Tykhonov provides smooth reconstructions with competitive error levels, while Total Variation achieves higher PSNR and markedly better SSIM, indicating superior preservation of edges and geometric structures.

4 Conclusion

Everything considered, it is possible to analyse the qualitative and quantitative differences between Tykhonov and Total Variation (TV) regularisation, in both denoising and deblurring tasks, especially in how they handle edges, smoothness, and computational time.

Tykhonov tends to produce very smooth images because it penalises large gradients. This makes the optimisation straightforward and fast. The downside is that fine details and sharp edges often get blurred.

TV, on the other hand, preserves edges and sharper features better. The images look closer to the original structure. However, the optimisation takes longer, and most important, **finding and selecting the right parameters requires more analysis**.

We selected the parameters by visually inspecting the results and evaluating metrics such as PSNR, SSIM, MSE, and MAE. In this study, we worked within the specific σ we considered, and the optimal regularisation strength reflected those conditions. In other scenarios with different noise characteristics or degradations, it would be important to revisit the parameter choices and adapt λ accordingly rather than using fixed values.

These patterns were consistent across both denoising and deblurring experiments, showing the practical changes between sharpness, smoothness, and speed.

5 Limitations and Future Work (Extended)

Due to time constraints, more advanced optimizers such as conjugate gradient, ADMM, or primal–dual methods were not implemented.

Future extensions could include:

- Blind deconvolution (unknown blur kernel)
- Spatially varying blur
- Learned priors via neural networks

Appendix

A Implementation Details

A.1 PyTorch Implementations

All gradient descent methods were implemented in PyTorch. Key implementation choices:

- Periodic boundary conditions implemented via FFT
- Double precision tensors for stability
- Manual gradient updates (no use of built-in PyTorch optimizers)

Pseudocode for Tykhonov GD:

```
u = noisy_image.clone()
for k in range(max_iter):
    grad = (u - v) - lambda * laplacian(u)
    u = u - lr * grad
```

Pseudocode for TV GD:

```
u = noisy_image.clone()
for k in range(max_iter):
    gx, gy = gradient(u)
    g = sqrt(gx**2 + gy**2 + eps**2)
    div = divergence(gx/g, gy/g)
    grad = (u - v) - lambda * div
    u = u - lr * grad
```

B Additional Experiments

B.1 Failure Cases

We observed:

- Oversmoothing for very large λ values
- Staircasing artifacts in TV for very small ϵ

These experiments motivated the parameter ranges used in the main report.

C Boundary and FFT Artifacts

The Fourier implementations assume periodic boundary conditions. This explains the small border inconsistencies observed when subtracting Fourier and PyTorch solutions.

We tested zero-padding as an alternative, but it introduced stronger edge attenuation. Periodic boundaries were therefore retained for consistency.

D Libraries and Reproducibility

All experiments were run using:

- Python 3.10
- PyTorch
- NumPy
- SciPy
- scikit-image (for PSNR, SSIM, MAE, MSE)

D.1 Reproducibility Settings

```
torch.manual_seed(0)
np.random.seed(0)
```

Image normalization:

$$u \in [0, 1]$$

was used consistently across all experiments.

E Extra Visual Results

E.1 Additional Qualitative Comparisons

In this section we present several complementary visual results to further illustrate the behaviour of the studied regularisation methods.

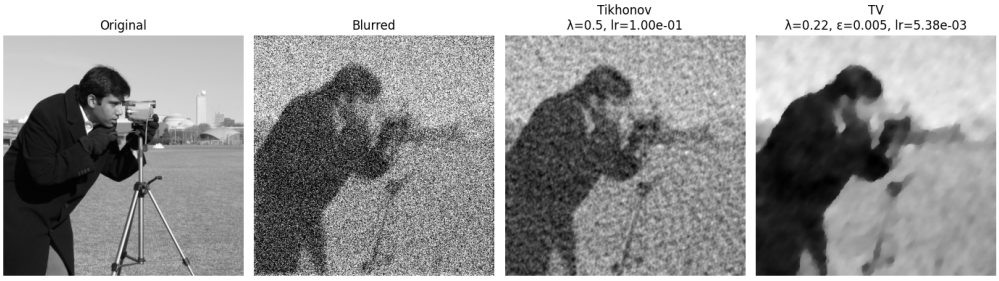


Figure 10: Denoising under strong Gaussian noise.

Analys : In this extreme degradation scenario, both methods struggle to restore meaningful structure. Tikhonov leaves a very large amount of residual noise, producing a reconstruction that remains highly grainy and lacks clear contours. Total Variation manages to suppress more noise and recovers coarse shapes more effectively, but the reconstruction is still heavily oversmoothed, with severe loss of detail and visible artefacts. Overall, neither method fully compensates for such extreme noise, but TV remains noticeably more robust than Tikhonov.

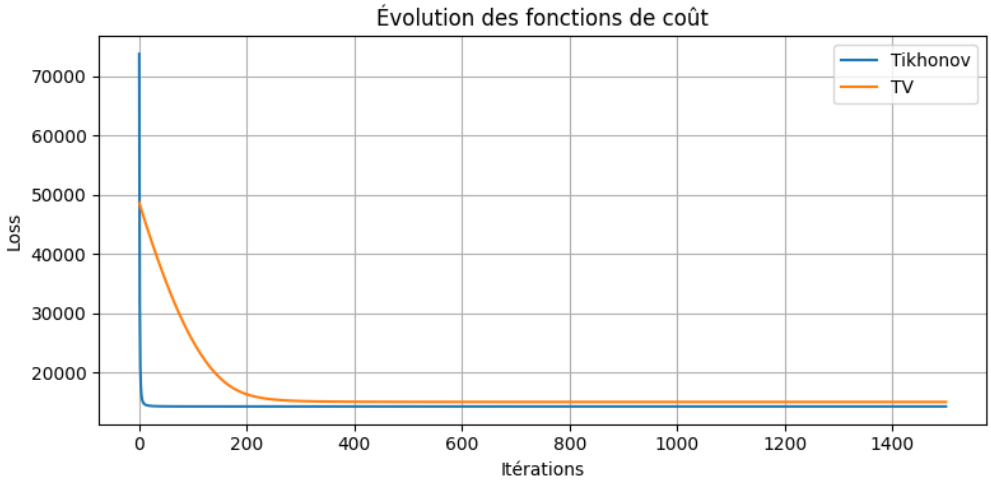


Figure 11: Cost function evolution under strong Gaussian noise.

Analysis : Even though the power of the noise have been drastically increased, the convergence of the cost function is roughly the same as the usual case.



Figure 12: Debluring using Total Variation method with different values of lambda and epsilon.

Description. This figure is used to analytically decided which value of lambda and epsilon are the best. Here, we are determining the parameters of the Total Variation reglarisation function, but the same method was used for all the other cases



Figure 13: Deblurring using Tikhonov method with different values of λ.