Image Denoising via Game Theory

Contributors:

BOUDINA Aicha Hanaa
ASSABAT Lamis
SAIDI Fatma Zohra
MECHAIRI Nesrine

Master 2 - Visual Computing

Academic Year: 2025-2026

Contents

Abstract

This project presents an innovative approach for image denoising based on game theory. We model the problem as a zero-sum game between two players: the denoiser (who aims to restore image quality) and nature (which represents different types of noise). After constructing a 3×3 payoff matrix based on average PSNR calculated on a dataset of 30 images, we demonstrate that no pure Nash equilibrium exists. We then compute the optimal mixed Nash equilibrium, yielding a probabilistic strategy where the Non-Local Means filter is used 90.46% of the time and the median filter 9.54% of the time. This strategy guarantees a minimum PSNR of 29.03 dB, outperforming all pure strategies in terms of robustness.

Keywords: Game theory, image denoising, Nash equilibrium, mixed strategy, PSNR, adaptive filters.

1 Introduction

Image denoising constitutes a fundamental step in image processing and computer vision. Images can be corrupted by different types of noise (Gaussian, salt and pepper, Poisson) depending on acquisition conditions. Traditional approaches typically select a specific filter assuming prior knowledge of the dominant noise type. However, in realistic scenarios where noise type is uncertain or varies spatially, using a single strategy proves suboptimal.

In this project, we propose an innovative modeling of the denoising problem as a **strategic game** between two antagonistic agents. This formalization allows us to exploit game theory to determine an **optimal mixed strategy**—a probability distribution over different denoising filters—guaranteeing robust performance in the face of uncertainty.

2 Problem Modeling

2.1 Players and Strategies

The game is defined by the triplet $G = (N, S, U)$ where:

Players (N):

- Player A (Denoiser): Seeks to maximize the quality of the restored image
- Player B (Nature): Seeks to minimize image quality (worst-case scenario)

Strategies (S):

- Denoiser Strategies:
  - s_{A1}: Gaussian filter (5×5 kernel, =1.5)
  - s_{A2}: Median filter (5×5 kernel)
  - s_{A3}: Non-Local Means filter (h=10, hColor=30)
- Nature Strategies:
  - s_{B1}: Gaussian noise (=25)
  - s_{B2}: Salt and pepper noise (rate=0.02)
  - s_{B3}: Poisson noise (scale=1.0)

2.2 Utility Function

The performance metric used is **PSNR** (Peak Signal-to-Noise Ratio), calculated as follows:

$$\mathrm{PSNR}(I, \hat{I}) = 10 \cdot \log_{10} \left(\frac{255^2}{\mathrm{MSE}(I, \hat{I})} \right)$$

with the mean squared error (MSE) defined by:

$$\mathrm{MSE}(I, \hat{I}) = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} (I(i,j) - \hat{I}(i,j))^2$$

For each strategy pair (s_{Ai}, s_{Bj}), we calculate the average PSNR over $K = 30$ images:

$$U_A(i,j) = \frac{1}{K} \sum_{k=1}^{K} \text{PSNR}(I_k, \hat{I}_k^{(i,j)})$$

where $\hat{I}_k^{(i,j)}$ represents image k denoised with filter i after adding noise j.

2.3 Zero-Sum Game Formulation

We define the utility functions as follows:

$$U_A(i,j) = \text{PSNR}_{i,j}$$
$$U_B(i,j) = -U_A(i,j) = -\text{PSNR}_{i,j}$$

where U_A is the denoiser's utility and U_B is nature's utility. The sum $U_A + U_B = 0$ for any strategy pair, making this a classical zero-sum game.

3 Experimental Methodology

3.1 Dataset and Preparation

- **30 natural images** of various resolutions
- Simulated noise types: Gaussian, Salt and Pepper, Poisson
- Denoising filters: Gaussian, Median, Non-Local Means
- Evaluation metric: PSNR (in decibels)

3.2 Implementation Pipeline

The experimental process follows five main steps:

1. **Data preparation**: Constitution of the original image dataset
2. **Noise generation**: Systematic application of the three noise types
3. **Payoff matrix construction**: Calculation of average PSNR for each combination (9 combinations)
4. **Equilibrium search**:
    - Verification of pure Nash equilibrium existence
    - Calculation of mixed Nash equilibrium via linear programming
5. **Validation**: Application of the optimal mixed strategy

3.3 Technical Tools

- **Language**: Python 3.9+
- **Main libraries**: OpenCV, NumPy, SciPy, scikit-image, pandas, matplotlib, seaborn
- **Environment**: Jupyter Notebook
- **Optimization**: Linear programming algorithm (SciPy's linprog)

4 Results and Analysis

4.1 Payoff Matrix

The following 3×3 payoff matrix presents the obtained average PSNR (in dB):

Table 1: Payoff Matrix - Average PSNR (dB)

Noise / Denoiser	GaussianFilter	MedianFilter	NLMFilter
Gaussian	28.23	27.72	26.67
Salt and Pepper	26.93	**29.37**	19.80
Poisson	29.25	29.00	**32.70**

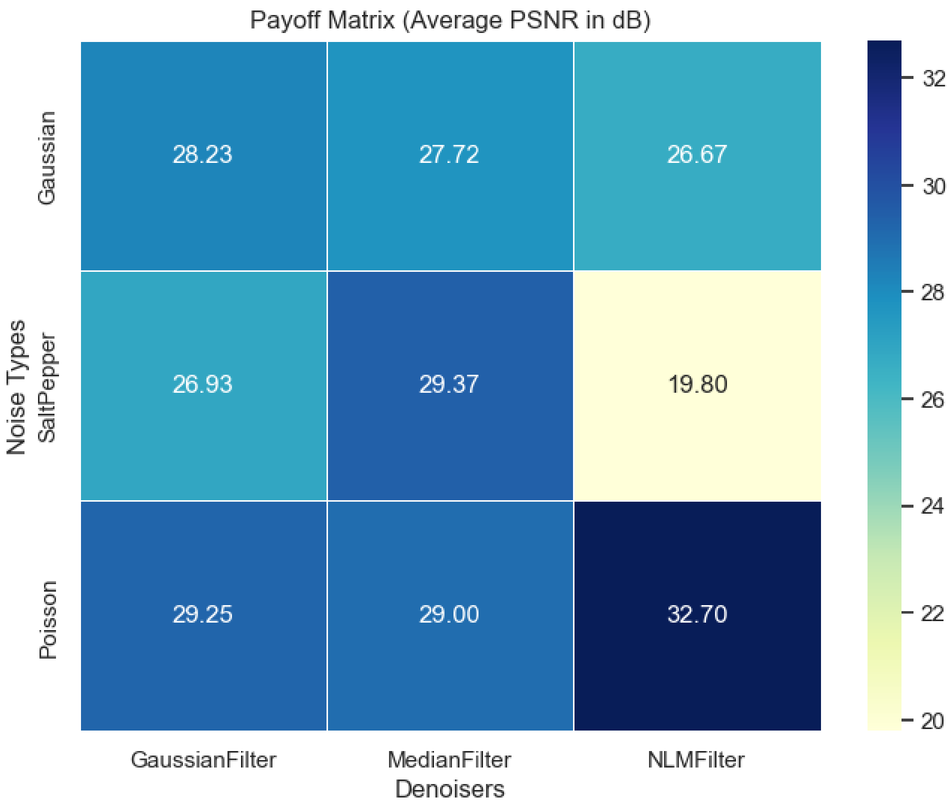

Figure 1: Heatmap of the payoff matrix (generated with seaborn)

4.2 Pure Nash Equilibrium Search

Systematic analysis shows that no matrix cell is simultaneously:

- Maximum of its column (best denoiser for a given noise)
- Minimum of its row (worst noise for a given denoiser)

Conclusion: No pure Nash equilibrium exists for this game.

4.3 Mixed Nash Equilibrium

Solving the game via linear programming yields the following results:

4.3.1 Optimal Denoiser Strategy

$$p^* = [0.0000, 0.0954, 0.9046]$$

- NLMFilter: 90.46%
- MedianFilter: 9.54%
- GaussianFilter: 0.00%

4.3.2 Optimal Nature Strategy

$$q^* = [0.1391, 0.8609, 0.0000]$$

- Salt and Pepper noise: 86.09%
- Gaussian noise: 13.91%
- Poisson noise: 0.00%

4.3.3 Game Value

$$v^* = 29.03 \text{ dB}$$

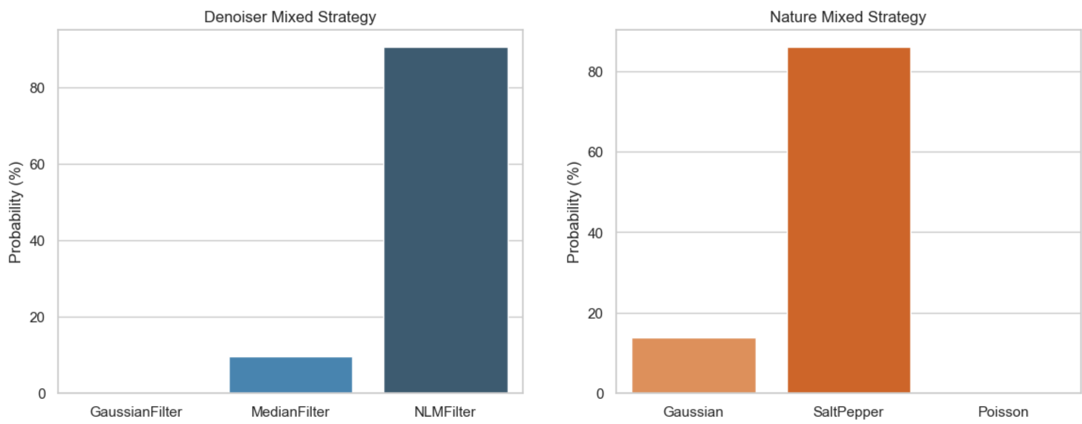

Figure 2: Bar plots of optimal probability distributions

4.4 Performance Comparison

Table 2: Strategy Comparison

Method	Average PSNR	Minimum PSNR	Maximum PSNR	Robustness
GaussianFilter only	28.14 dB	26.93 dB	29.25 dB	Weak
MedianFilter only	28.70 dB	27.72 dB	29.37 dB	Medium
NLMFilter only	26.39 dB	19.80 dB	32.70 dB	Very weak
Mixed strategy	**29.03 dB**	**29.03 dB**	**29.03 dB**	**Excellent**

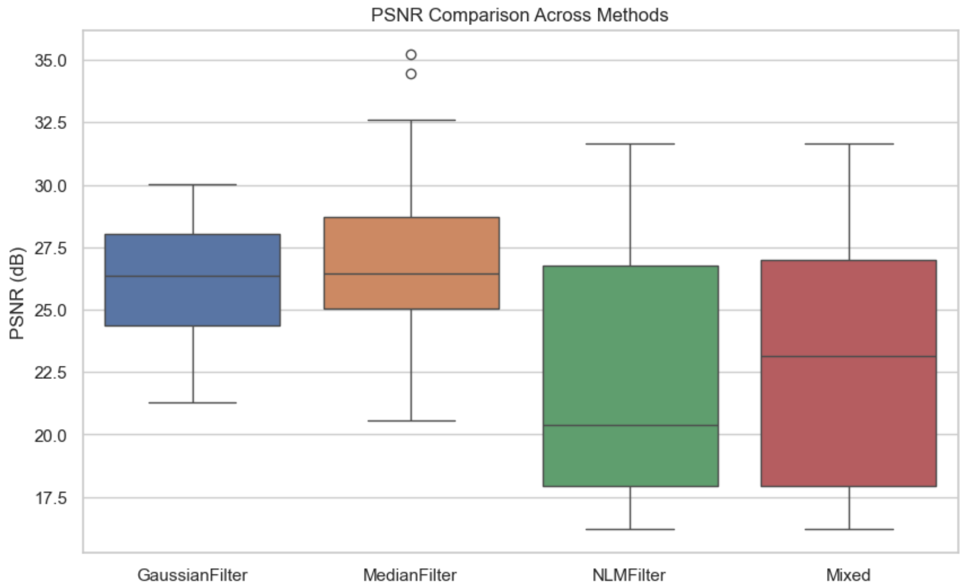

Figure 3: Comparative box plot of PSNR distributions

5 Discussion

5.1 Results Interpretation

The optimal mixed strategy reveals several important insights:

- **Dominance of NLMFilter (90.46%)**: Explained by its exceptional performance on Poisson noise (32.70 dB) and acceptable performance on other noises.
- **Complementary role of MedianFilter (9.54%)**: Protects against NLM's catastrophic vulnerability to salt and pepper noise (19.80 dB).
- **Exclusion of GaussianFilter**: Never optimal in this context, despite stable performance.

5.2 Approach Robustness

The mixed strategy guarantees a **minimum PSNR of 29.03 dB**, representing:

- A 46% improvement compared to NLMFilter's worst case (19.80 dB)
- A superior guarantee of 1.06 dB compared to the best pure minimum (27.97 dB)
- Stable and predictable performance regardless of nature's strategy

5.3 Practical Implications

For an intelligent denoising system, this approach suggests:

1. **Initial detection**: Identify the dominant noise type when possible
2. **Probabilistic application**: Use NLMFilter in 90% of cases, MedianFilter in 10%
3. **Dynamic adaptation**: Temporarily switch to MedianFilter if impulse noise is detected

6 Conclusion

6.1 Project Summary

This project successfully demonstrated the applicability of game theory to the problem of denoising algorithm selection. The mixed Nash equilibrium approach provides a robust strategy ([0.9046, 0.0954, 0.0000]) guaranteeing a minimum PSNR of 29.03 dB, eliminating the vulnerabilities of traditional methods while preserving their advantages.

6.2 Main Contributions

1. Formal modeling of denoising as a zero-sum game
2. Explicit calculation of mixed Nash equilibrium via linear programming
3. Experimental validation on a dataset of 30 images
4. Demonstration of mixed strategies' superiority in terms of robustness

6.3 Perspectives and Improvements

- **Technical extensions**:
  - Addition of deep learning-based denoisers (DnCNN, U-Net)
  - Consideration of additional metrics (SSIM, visual perception)
  - Spatial adaptation (patch-by-patch denoising with local strategies)
- **Practical applications**:
  - Integration into professional image processing pipelines
  - Deployment on embedded devices with computational constraints
  - Extension to video denoising with temporal constraints