Data-driven diffusion-based super-resolution applied to Rayleigh-Bénard convection

Ruan F. S. Sousa[1], Roberto M. Velho[1], Gabriel F. Barros[2], Adriano M. A. Cortes[1,4], Fernando A. Rochinha[3], Alvaro L. G. A. Coutinho[2]

[1]*High Performance Computing Center and Systems Engineering and Computer Science, COPPE/Federal University of Rio de Janeiro*
Av. Horácio Macedo 2030 CT, Bloco H - 319, 21941-914, Rio de Janeiro, Brazil
ruanfelipe@cos.ufrj.br, robertovelho@cos.ufrj.br, adricortes@cos.ufrj.br
[2]*High Performance Computing Center and Civil Engineering, COPPE/Federal University of Rio de Janeiro*
Av. Athos da Silveira Ramos, 149 CT, Bloco B - 101, 21941-909, Rio de Janeiro, Brazil
gabriel.barros@coc.ufrj.br, alvaro@coc.ufrj.br
[3]*Mechanical Engineering, COPPE/Federal University of Rio de Janeiro*
Av. Horácio de Macedo, 2030 CT, Bloco G - 204, 21941-914, Rio de Janeiro, Brazil
faro@mecanica.ufrj.br
[4]*Applied Mathematics Department Mathematics Institute/Federal University of Rio de Janeiro*

Abstract. In computational fluid dynamics, numerical simulations produce reliable and accurate results but incur a high time and computational cost investment. Numerical data-driven models tend to be more computationally efficient but lack the accuracy of numerical simulations and, in some cases, present numerical phantoms, small fluctuations caused by numerical issues. We propose using two modern machine learning techniques in a sequence: a surrogate model for the fluid flow coupled to a super-resolution diffusion probabilistic model. The second model aims to improve the accuracy of predictions produced by the first one. We described a simplified version of what those models could produce together, exemplified by the Rayleigh-Bénard convection problem. The experiments show that this technique works for denoising and the downscaling case examples.

Keywords: Super-resolution, Generative Models, Diffusion probabilistic model, Rayleigh-Bénard problem

1 Introduction

Super-resolution (SR) techniques target the improvement of images that can only be obtained with lower quality for many reasons. Surrogate models aim to reproduce the dynamics of a certain physical system while eventually incurring a certain level of error. Their benefit is the reduced computational complexity, which makes them faster than the computation of the full high-fidelity dynamics. Modern machine learning techniques for surrogates of fluid flows have shown great performance, see Vinuesa and Brunton [1].

Data for training such surrogates may be low-fidelity due to the offline computational cost. What also tends to produce low-fidelity predictions. Thus, different techniques may be employed to improve the quality of such predictions. A recently proposed one is Shu et al. [2], whose goal is to reconstruct high-fidelity images (akin to images produced by Direct Numerical Simulation - DNS) from low-fidelity predictions of turbulent fluid flows produced by techniques such as Large Eddy Simulations (LES), Reynolds Averaged Navier-Stokes (RANS), their combination RANS-LES, or functional sub-grid models.

We take a different approach, proposing using surrogate models produced by machine learning coupled with a super-resolution technique adapted from Shu et al. [2]. Our primary interest is in surrogate models for parametric models of fluid flows (eventually described by systems of partial differential equations - PDEs). Our surrogate is trained similarly to the one described in Guo and Hesthaven [3], Fresca and Manzoni [4]. The goal is to be able to reproduce the dynamics of a fluid flow driven by the system of equations of interest for an unseen (in the training stage) parameter. Our goal is to couple the predictions produced by the surrogate model as input for the already trained super-resolution technique, aiming to produce images with higher accuracy when compared to their true value (test data) than just the surrogate prediction. The super-resolution machine learning model must also be trained with the same data as the surrogate to achieve this goal. The full description of the surrogate model

construction can be found in Rochinha et al. [5] or in a preprint from part of authors of this current work, Cortes et al. [6]. In the present work, we describe a simpler version that uses the test data selected for the surrogate model as input test data for the super-resolution model instead of generated predictions from the test data. This simplification speeds up the implementation of a computational code to assess the SR models' efficiency while reducing the complexity of the description of both techniques, thus focusing on the essential aspects of the SR proposed usage.

In order to exemplify the method, we choose the Rayleigh-Bénard (RB) problem. For a recent review on the challenges of this important thermo-convection problem see Song et al. [7]. We construct two datasets for training and testing the associated surrogate model and we use such datasets for training and testing the model that performs the image resolution improvement. On a future work, in preparation, we intend to describe in details the full system of surrogate and super-resolution models coupled, showing the accuracy at each stage, i.e., the produced predictions from the surrogate and their reconstructions via the SR technique.

2 Methodology

In this section, we describe the RB problem formulation, the process of generating data for training and testing the SR model, the super-resolution model's theoretical aspects and its application.

2.1 Problem Setup

The Rayleigh-Bénard (RB) convection problem is defined here as the coupling of the incompressible Navier-Stokes and the energy (heat) equations, Griebel et al. [8]. Using the dimensionless variables,

$$\mathbf{x}^* = \frac{\mathbf{x}}{L}, \quad t^* = \frac{t}{L/U_0}, \quad \mathbf{u}^* = \frac{\mathbf{u}}{U_0}, \quad p^* = \frac{p}{\rho\, {U_0}^2}, \quad \theta^* = \frac{g\beta\Delta T L^3}{\nu^2}, \tag{1}$$

with $\mathbf{x}$ the Cartesian coordinates, t time, $\mathbf{u}$ the velocity, p the pressure, $\Delta T = T - T_0$ the temperature difference (assuming T_0 as the reference temperature at the top of the domain), U_0 the reference velocity, L the reference length, g the gravitational acceleration, ρ the density, β the thermal expansion coefficient, and ν is the kinematic viscosity. From now on, we will drop the superscript $*$ in the equations. Therefore, the system of equations is written in dimensionless form as

$$\nabla \cdot \mathbf{u} = 0, \tag{2a}$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \sqrt{\frac{Pr}{Ra}}\nabla^2 \mathbf{u} + \theta\, \mathbf{e}_2, \tag{2b}$$

$$\frac{\partial \theta}{\partial t} + (\mathbf{u} \cdot \nabla)\theta = \frac{1}{\sqrt{Ra \cdot Pr}}\nabla^2 \theta, \tag{2c}$$

where $\mathbf{e}_2$ is the canonical unitary vector in y direction, eq. (2a) is the continuity equation, eq. (2b) is the momentum equation, and eq. (2c) describes the energy equation in terms of the thermal buoyancy. All the fields are functions of $(t, \mathbf{x})$, with $\mathbf{x} \in \Omega \subset \mathbb{R}^2$. The dimensionless numbers

$$Pr = \frac{\nu}{\alpha}, \quad Ra = \frac{g\beta\Delta T L^3}{\nu\alpha} \tag{3}$$

represent the Prandtl and Rayleigh numbers, respectively, with α being the thermal diffusivity. Regarding the boundary conditions, for a square domain Ω, we have no-slip for $\mathbf{u}$ and prescribed θ on top and bottom and periodic conditions in the horizontal direction. The initial condition is zero velocity in the whole domain, and a constant gradient energy profile pointing downward is added with a small seeded Gaussian random disturbance to trigger the instability.

2.2 Dataset Generation and Processing

The RB governing equations are discretized by a spectral method using the open-source library Dedalus, Burns et al. [9]. The domain Ω has dimensions 1 by 1, with 256^2 grid points, and we select a Fourier basis in the x direction and a Chebyshev basis in the y direction. All simulations are performed from time 0 to 80 with a fixed time step of 1×10^{-3} with a 2nd-order, 2-stage, Diagonally Implicit Runge–Kutta (DIRK) method, for $Pr = 0.72$ and $Ra = 5 \times 10^6$. Details on how to solve the RB problem numerically are found in Griebel et al. [8]. Snapshots of the dynamics are taken every 0.25 time instants, thus generating 321 snapshots per simulation. As outputs, both buoyancy and vorticity are calculated, but we selected only buoyancy data to be used in the SR technique. In order to generate sufficient training data, we employed 20 different seeds on the Gaussian random perturbation that triggers the instability, generating completely different dynamics for each seed. Additionally, three seeds are used to generate test data.

Each time instant of the dynamics corresponding to an image of 256 by 256 grid points is reshaped to a vector of dimension 65536. Thus, a description of the dynamics for each simulation (each seed) is a matrix of dimension 65536 (corresponding to space) by 321 (corresponding to time) - organized in temporal order. We then construct a large snapshot matrix, as in the surrogate model's algorithm. Such a matrix is the concatenation of all matrices previously described, i.e., we concatenate the dynamics corresponding to all seeds in a single large matrix in the same order in which they are constructed. This large snapshot matrix is the training dataset for the surrogate model and also the super-resolution one. The same approach can be used to assemble a test dataset from concatenating matrices corresponding to the dynamics associated with test seeds. We are then ready to train the super-resolution model.

2.3 Mathematical Description of the Super-resolution method

The super-resolution method here described is based on a denoising diffusion probabilistic model (DDPM). A DDPM is a Markov chain-based generative model that, starting from a sample of a standard Gaussian distribution, reconstructs an original sample. The model can reverse a Gaussian diffusion process by progressively estimating a transition function p_θ, where θ are learnable neural network parameters. Let x_0 be the data sample to be reconstructed, that is, the final result. The reconstruction procedure, also called a *backward diffusion process*, can be represented as:

$$p_\theta\left(x_{0:T}\right) = p(x_T) \prod_{t=1}^{T} p_\theta\left(x_{t-1}|x_t\right), \tag{4}$$

where T is a large enough step for the sample to be reduced to a standard Gaussian distribution in the forward process, and with $x_T = \mathcal{N}(x_T; 0, I)$, where $\mathcal{N}(x; \mu, \beta I)$ is a Gaussian kernel with mean μ and variance βI, I being the identity matrix. The probability transition $p_\theta\left(x_{t-1}|x_t\right)$ is chosen as:

$$p_\theta\left(x_{t-1}|x_t\right) = \mathcal{N}\left(x_{t-1}; \mu_\theta\left(x_t, t\right), \Sigma_\theta\left(x_t, t\right)\right). \tag{5}$$

The transition functions μ_θ and Σ_θ are learned through the training process, in which given a high fidelity image x_{start}, Gaussian noise is progressively added to the data until it is reduced into random noise. This process is called a *forward diffusion process*. Starting from a high-fidelity sample, the forward process can be constructed as follows:

$$q\left(x_{1:T}|x_{start}\right) = \prod_{t=1}^{T} q\left(x_t|x_{t-1}\right), \quad q\left(x_t|x_{t-1}\right) = \mathcal{N}\left(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I\right), \tag{6}$$

where $\beta_1, \ldots, \beta_T$ is a progressive variance scaling factor; and q is a probability transition function. From the final sample of this process, a reconstruction process is attempted, the loss is evaluated, and the parameters in θ are updated. In principle, the loss function to be minimized is the negative log-likelihood, $-\log p_\theta(x_{start})$, which is computationally intensive, and whose gradient can be hard to calculate. Instead, we follow the work of Ho and Salimans [10], showing one can instead minimize the following function:

$$L_t^{simple} = \mathbb{E}_{t\sim[1,T],x_{start},\epsilon}\left[\left\|\epsilon_t - \epsilon_\theta\left(\sqrt{\bar{\alpha}_t}\,x_{start} + \sqrt{1-\bar{\alpha}_t}\,\epsilon_t, t\right)\right\|^2\right], \tag{7}$$

where $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, $\epsilon_t \sim \mathcal{N}(0, I)$ is the standard Gaussian noise sampled at time t, and $\epsilon_\theta(\cdot,\cdot)$ is the reconstruction of the model, given $\sqrt{\bar{\alpha}_t}\,x_{start} + \sqrt{1-\bar{\alpha}_t}\,\epsilon_t$ and t as inputs.

2.4 Application of the Technique

The diffusion method described in section 2.3 is then applied to the training data, following the steps in Shu et al. [2], but without the guided diffusion procedure. For evaluation, we execute experiments applying the trained diffusion super-resolution model on out-of-sample test data of three forms: downscaling, Gaussian denoising, and Salt-and-pepper denoising, described in what follows. Moreover, remember the description of buoyancy in section 2.1 whose values are between 0 and 1. Given the colormap chosen to visualize the figures in this work, pixels representing buoyancy around 0 are close to blue, while the ones around 1 are close to orange.

In the downscaling case, we first reduce the resolution of the original test sample 16 times (from 256 by 256 to 16 by 16). Then, we apply nearest-neighbour interpolation to upscale back to the original size. We apply the super-resolution diffusion model over such images that have passed through upscaling. For the Gaussian denoising case, we add a different realization of Gaussian noise K to every image, $K \sim \mathcal{N}(0, 0.5 * I)$, then reconstruct them via the model.

Finally, for Salt-and-pepper denoising case, the "salt-and-pepper" noise works as follows: for each pixel of an image, a random number between 0 and 1 is draw with uniform probability. If its value is below 0.05, the pixel is redefined as 0. If its value is above 0.95, the pixel is redefined as 1. For any other value, the pixel is not modified. As in the other two cases, after the images in the test dataset suffer from this noise, we reconstruct them via the diffusion model. We observe the model's results over such three cases in section 3. The training of the super-resolution model is performed with an NVIDIA A400 GPU, along 300 epochs, for about 16 hours. We made use of the code in Shu et al. [11] associated to Shu et al. [2], written in the PyTorch framework, applying the ADAM optimizer, with a constant learning rate of 2×10^{-4}, and batch size of 12.

3 Results

We conducted three experiments described in section 2.4 to evaluate the model's accuracy. For each experiment, we pick a different set of 20 images, corresponding to 20 contiguous time instants in sequential order of the dynamics. We show the ground truth, the input image - corresponding to the noisy or down-scaled version of the corresponding ground truth, the reconstruction via the super-resolution model, and the absolute error between the reconstruction and the ground truth.

For the downscaling case, in which the method softens and hides the original information, results can be seen in Fig. 1. We observe that the model can reconstruct the original dynamics. More significant errors in the reconstruction occur in colder regions of the dynamics. For the second experiment, model can perform images' denoising, but it produces larger errors for the buoyancy on the top and bottom of the domain, where boundary conditions are applied. The results can be seen in Fig. 2.In the third experiment, with the added salt-and-pepper noise, the model is also to reconstruct the original dynamics, and again, larger errors are seen in the top and bottom parts of the domain, see Fig. 3.

4 Conclusions

We have shown that a diffusion-based super-resolution model can reconstruct the original dynamics from noisy images or the ones passed through downscaling. Although non-negligible errors may occur, most where the boundary conditions are applied, using such a technique in conjunction with a surrogate model is promising. First, because the level of disturbance predictions via surrogate models produce, see Cortes et al. [6], seem to be lower than the disturbances we used in this current work. Second, the issue of boundary conditions may be better treated by introducing a residual-based physics-informed super-resolution technique, which is the subject of ongoing work.

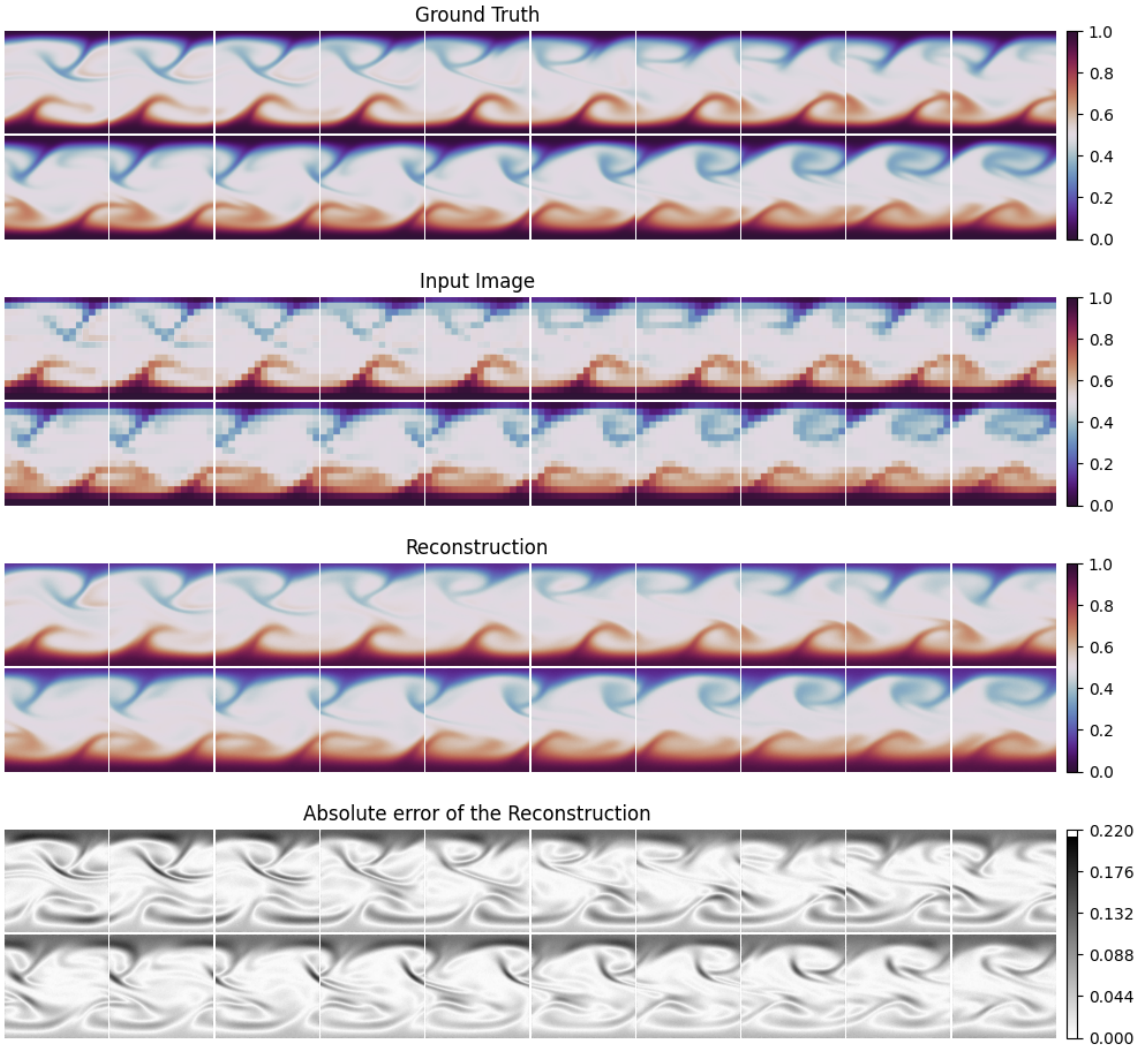

Figure 1. Rayleigh-Bénard dynamics associated to the downscaling case.

Acknowledgements. This study was financed in part by CAPES, Brazil Finance Code 001. This work is also partially supported by FAPERJ, CNPq, and Petrobras.

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] R. Vinuesa and S. L. Brunton. Enhancing computational fluid dynamics with machine learning. *Nature Computational Science*, vol. 2, n. 6, pp. 358–366, 2022.
[2] D. Shu, Z. Li, and A. Barati Farimani. A physics-informed diffusion model for high-fidelity flow field reconstruction. *Journal of Computational Physics*, vol. 478, pp. 111972, 2023.
[3] M. Guo and J. S. Hesthaven. Data-driven reduced order modeling for time-dependent problems. *Computer Methods in Applied Mechanics and Engineering*, vol. 345, 2018.
[4] S. Fresca and A. Manzoni. Pod-dl-rom: Enhancing deep learning-based reduced order models for nonlinear parametrized pdes by proper orthogonal decomposition. *Computer Methods in Applied Mechanics and Engineering*, vol. 388, 2022.
[5] F. Rochinha, A. Côrtes, R. Velho, A. Coutinho, G. Guerra, R. Elias, J. Camata, E. Santos, and G. F. Barros. Combining high-performance computing with machine learning to model turbidity currents. *AGU23*, 2024.

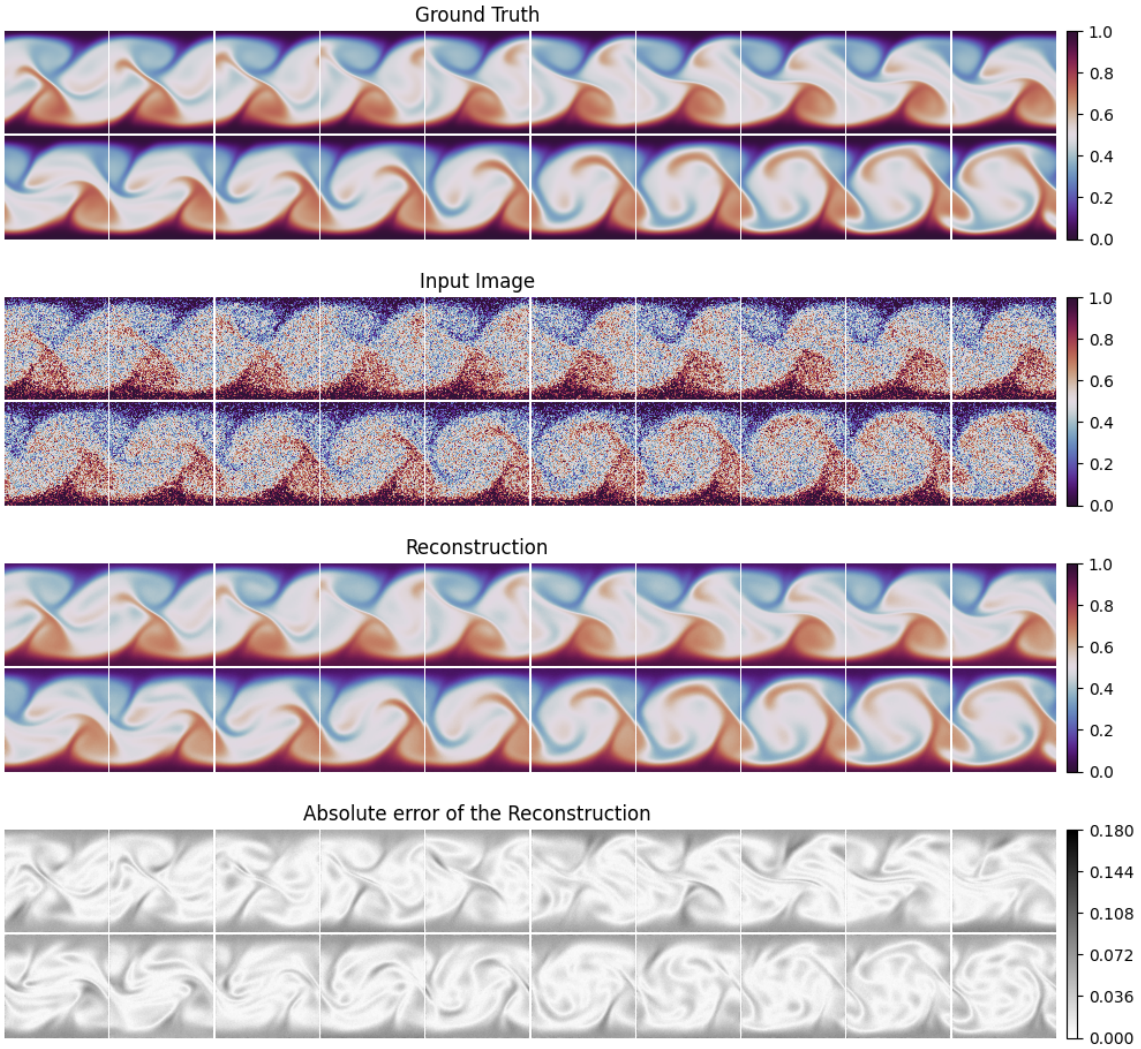

Figure 2. Rayleigh-Bénard dynamics associated to the Gaussian noise case.

[6] A. Cortes, R. Velho, G. Barros, F. Rochinha, and A. Coutinho. Ablation study of a non-intrusive data-driven surrogate model for the rayleigh-benard convection problem. *CILAMCE 2024 Preprint*, 2024.

[7] J.-J. Song, P.-X. Li, L. Chen, C.-H. Li, B.-W. Li, and L.-Y. Huang. A review on rayleigh-bénard convection influenced by the complicating factors. *International Communications in Heat and Mass Transfer*, vol. 144, pp. 106784, 2023.

[8] M. Griebel, T. Dornseifer, and T. Neunhoeffer. *Numerical simulation in fluid dynamics: a practical introduction*. Society for Industrial and Applied Mathematics, USA, 1998.

[9] K. J. Burns, G. M. Vasil, J. S. Oishi, D. Lecoanet, and B. P. Brown. Dedalus: A flexible framework for numerical simulations with spectral methods. *Physical Review Research*, vol. 2, n. 2, pp. 023068, 2020.

[10] J. Ho and T. Salimans. Classifier-free diffusion guidance. *arXiv:2207.12598*, 2022.

[11] D. Shu, Z. Li, and A. B. Farimani. Diffusion-based-fluid-super-resolution. `https://github.com/BaratiLab/Diffusion-based-Fluid-Super-resolution/tree/main_v1`, 2023.

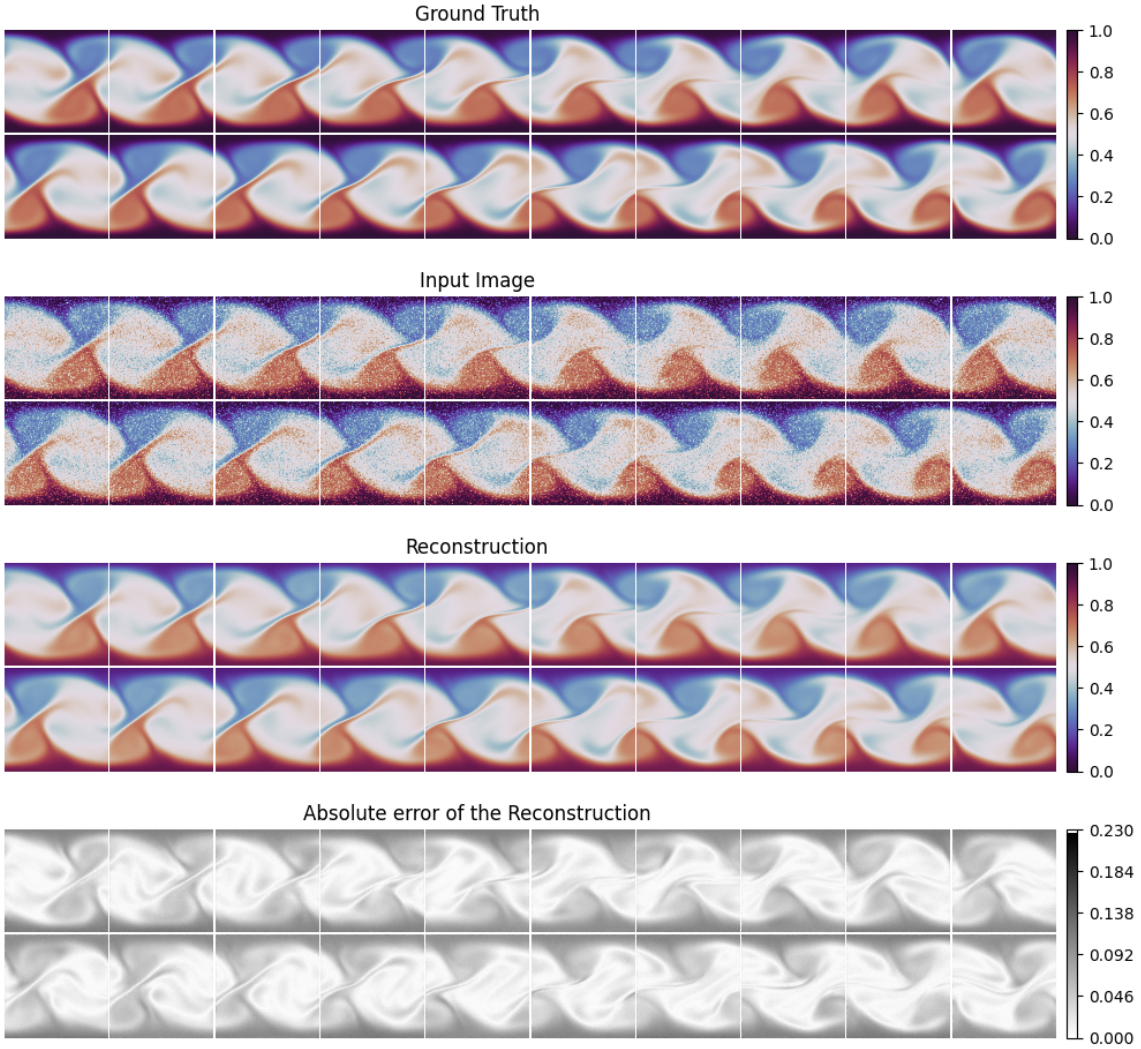

Figure 3. Rayleigh-Bénard dynamics associated to the salt-and-pepper noise case.