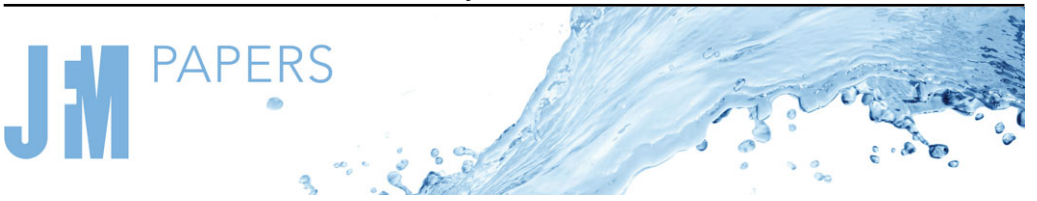
J. Fluid Mech. (2025), *vol.* 1006, A10, doi:10.1017/jfm.2025.21



Reduced representations of Rayleigh–Bénard flows via autoencoders

Melisa Y. Vinograd[1,2] **and Patricio Clark Di Leoni**[1,3]

[1]Departamento de Ingeniería, Universidad de San Andrés, Buenos Aires, Victoria 1644, Argentina

[2]Departamento de Física, Universidad de Buenos Aires, CABA 1868, Argentina

[3]CONICET, CABA 1414, Argentina

Corresponding author: Patricio Clark Di Leoni, pclarkdileoni@udesa.edu.ar

(Received 22 May 2024; revised 13 November 2024; accepted 21 December 2024)

We analysed the performance of convolutional autoencoders in generating reduced-order representations of the temperature field of two-dimensional Rayleigh–Bénard flows at $Pr = 1$ and Rayleigh numbers extending from 10^6 to 10^8, capturing the range where the flow transitions to turbulence. We present a way of estimating the minimum number of dimensions needed by the autoencoders to capture all the relevant physical scales of the data that is more apt for highly multiscale flows than previous criteria applied to lower-dimensional systems. We compare our architecture with two regularized variants as well as with linear methods, and find that manually fixing the dimension of the latent space produces the best results. We show how the estimated minimum dimension presents a sharp increase around $Ra \sim 10^7$, when the flow starts to transition to turbulence. Furthermore, we show how this dimension does not follow the same scaling as the physically relevant scales, such as the dissipation length scale and the thermal boundary layer.

Key words: machine learning, turbulent convection, turbulent transition

1. Introduction

Generating reduced-order representations of turbulent flows is a major challenge underpinning several tasks and goals in fluid dynamics. They are essential both for modelling, as when writing reduced-order models, and for analysis and theory, as, for example, in the quest to determine the existence, dimensions and characteristics of inertial manifolds. Thus, the list of techniques for dimensionality reduction and their associated history is long, going from Fourier decompositions and principal component analysis (PCA), to wavelets, and lately to neural network-based autoencoders (Brunton & Kutz 2019). While the former employ linear techniques to derive orthogonal bases to best

© The Author(s), 2025. Published by Cambridge University Press.

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

represent the data, the latter perform nonlinear decompositions and have been shown to be significantly more effective at generating compact representations of different datasets (Hinton & Salakhutdinov 2006; Goodfellow, Bengio & Courville 2016).

Fuelled by their achievements in various machine learning tasks, autoencoders have been applied to various problems in fluid mechanics. In the context of the Kuramoto–Shivashinky equation, Linot & Graham (2020, 2022) employed a combination of PCA and autoencoders to parameterize the inertial manifold of the equation, where the dynamics on the parametrized space were later modelled with neural ordinary differential equations (ODEs) (Chen *et al.* 2019). These processes served to estimate the dimension of the inertial manifold, matching the estimates obtained via analysis of unstable periodic orbits (UPOs) (Ding *et al.* 2016). Expanding on these methods, Floryan & Graham (2022) incorporated elements of manifold theory into their analysis of the latent space. Working with the Kolmogorov flow, Page *et al.* (2021, 2023) used autoencoders, along with a Fourier decomposition of the latent space, to find and characterize UPOs and relate them to bursting events. Similarly, Linot & Graham (2023), De Jesús & Graham (2023) and Constante-Amores, Linot & Graham (2024) used models of the latent space dynamics to search for UPOs in the flow in Kolmogorov, channel and pipe flows, respectively. Fukami, Nakamura & Fukagata (2020) and Eivazi *et al.* (2022) were able to determine a hierarchy of modes within latent space and facilitated the proposal of nonlinear mode decomposition of flows in fewer modes than their linear counterparts. In the realm of turbulent stratified flows, Foldes, Camporeale & Marino (2024) were able to improve the autoencoders capabilities to capture high-order statistics by adding extra penalizations to the loss functions.

In order to avoid the costly parameter sweeps involved in determining the smallest dimension needed for the autoencoder to represent the data properly, different regularized autoencoder variants have been proposed. Zeng & Graham (2023) utilized the implicit regularization method, first introduced by Jing, Zbontar & lecun (2020), to Kolmogorov flow, showing that, in their examples, the regularized variant was able to give the same estimate for the minimal numbers of modes as the non-regularized version. Sondak & Protopapas (2021) introduced L_1 regularization to the autoencoder model so as to create a sparse version of the latent vector. They successfully identified the optimal reduced latent space for the Kuramoto–Shivashinkyy equation while demonstrating the absence of a corresponding reduced latent space for the Korteweg–de Vries equation, reflecting its inherently infinite-dimensional dynamics. These regularized variants have not yet been applied to flows with large-scale separation.

As was mentioned previously and as many of the examples above show, autoencoders serve as a powerful tool to generate parametrizations of inertial manifolds (or, at the very least, parametrizations of submanifolds that capture the relevant dynamics) and estimate their dimensions. Inertial manifolds are finite-dimensional manifolds that contain the global attractor of nonlinear dissipative systems (Hopf 1948; Foias, Sell & Temam 1988), offering a simplified description through a finite set of ODEs (Temam 1990). This concept has been successfully applied to systems such as the Kuramoto–Sivashinsky equation, where the existence of such manifolds has been rigorously established (Hyman & Nicolaenko 1986). While this is not the case for Navier–Stokes flows (Temam 1990), several studies provide evidence of its possible existence (Chandler & Kerswell 2013; Willis, Short & Cvitanović 2016; Crowley *et al.* 2022; De Jesús & Graham 2023).

The main aim of this paper is to generate low-order representations of the temperature field of Rayleigh–Bénard flows and use them to estimate the dimensionality of the flow as it transitions to turbulence, with an added emphasis on understanding the performance of different autoencoders when dealing with strongly turbulent and multiscale data.

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

Heat-induced currents play a crucial role in various geophysical and industrial activities. For instance, in the geophysical context, these currents are essential for the circulation of the Earth's atmosphere (Hartmann, Moy & Fu 2001) and oceans (Våge *et al.* 2018). The presence of buoyancy enables the generation of convective rolls, large-scale structures that transport heat and matter from hot to cold regions of the fluid. The balance between buoyant and viscous forces weighted by the Prandtl number, which is the ratio of viscosity to thermal diffusivity, is measured by the Rayleigh number. As this balance changes, so does the behaviour of the flow. We focus on Rayleigh numbers ranging from 10^6 to 10^8, where turbulence starts to develop and the system experiences a sudden change in its properties, such as the emergence of multi-stability and long transient states (van der Poel *et al.* 2012), the reversal of large-scale structures (Sugiyama *et al.* 2010; Castillo-Castellanos *et al.* 2019), the appearance of new optimal steady states (Kooloth, Sondak & Smith 2021) and a loss of synchronization capabilities (Agasthya *et al.* 2022). These changes are associated with an increase in the number of structures the flow exhibits, which, in turn, indicate that the dimension of the inertial manifold of the system is increasing during the transition. The use of convolutional autoencoders to generate reduced-order models of Rayleigh–Bénard flows has been explored for $Ra = 10^7$ in Pandey *et al.* (2022), where the local convective heat flux, a central measure for characterizing mean turbulent heat transfer, was forecasted by reducing the datasets to a dimension of 40.

This paper is organized as follows. Section 2 details the theoretical background and the analytical methods employed, first introducing important aspects of Rayleigh–Bénard flows in § 2.1. This is followed by a detailed explanation of the different autoencoder architectures used in our study, § 2.2 outlines the design and functionality of the fixed-d autoencoder (FdAE) networks with a fixed latent space dimension, and § 2.3 introduces two modifications to the network to improve training times: implicit rank minimizing autoencoders (IRMAEs) and sparsity-inducing autoencoders (SIAEs). In § 2.4 we outline the empirical framework used to test our theories, the details of the numerical simulations and training of the neural networks. Results and conclusions are presented in §§ 3 and 4, respectively.

2. Problem set-up and methodology

2.1. *Rayleigh–Bénard flow*

Rayleigh–Bénard convection describes the motion of a planar cell of fluid flow that is heated from below. Under the Boussinesq approximation, density variations are assumed to be small and the velocity and temperature fields are coupled through the buoyancy term. For a rectangular domain of horizontal length L and height h, the equations take the form

$$\frac{\partial \boldsymbol{v}}{\partial t} + (\boldsymbol{v}\cdot\nabla)\boldsymbol{v} = \boldsymbol{\nabla} p + \nu\nabla^2\boldsymbol{v} - \alpha g T\hat{\mathbf{z}}, \tag{2.1}$$

$$\boldsymbol{\nabla}\cdot\boldsymbol{v} = 0, \tag{2.2}$$

$$\frac{\partial T}{\partial t} + (\boldsymbol{v}\cdot\boldsymbol{\nabla})T = \kappa\nabla^2 T, \tag{2.3}$$

where $\boldsymbol{v}$ is the velocity field, p the pressure, T the temperature, ν the kinematic viscosity, κ the thermal conductivity, $\hat{\boldsymbol{z}}$ the direction parallel to gravity and g the acceleration of gravity. For reasons outlined below, we work with the rescaled temperature fluctuations $\theta = (T - T_0)\sqrt{\alpha g h/\Delta T}$, with $T_0 = T_b - z\Delta T/h$ the linear background profile and T_b the

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

temperature of the bottom plate. We assume periodic boundary conditions in the horizontal direction and non-slip boundary conditions in the vertical direction throughout the whole work.

The dimensionless Rayleigh number, Ra, quantifies the balance between buoyancy forces and the combined effects of thermal diffusion and viscous dissipation and is given by

$$Ra = \frac{\alpha g h^3 \Delta T}{\nu \kappa}. \tag{2.4}$$

As Ra increases, the fluid undergoes several transitions: from stable stratification, characterized by conductive heat transfer, to the emergence of convective rolls or cells (Bénard cells), and eventually to chaotic and turbulent motion. At $Ra \approx 1700$ the fluid transitions from a conductive to convective behaviour. This occurs as linear instability mechanisms break the translational invariance of the flow leading to the appearance of convective rolls. We focus on Ra values ranging from 10^6 to 10^8. At these levels, the dynamics is highly convective, with rolls starting to break down and the flow showing clear signs of turbulence. A subsequent regime, not covered in this work, occurring when Ra reaches 10^{13}, is the so-called ultimate regime. Here, small-scale structures dominate, and rolls are no longer evident (Zhu *et al.* 2018).

The two other dimensionless numbers that characterize Rayleigh–Bénard convection are the Prandtl number $Pr = \nu/\kappa$, which we set to 1 in this study, and the Nusselt number Nu,

$$Nu(z) = \frac{\langle v_z T - \kappa \partial_z T \rangle_{x,t}}{\Delta T \, \kappa / h}, \tag{2.5}$$

where $\langle \cdot \rangle x, t$ is the time and spatial average at a height z. The Nusselt number compares the ratio of vertical heat transfer due to convection to the vertical heat transfer due to conduction. On account of energy conservation, the Nusselt number is constant at each height z when using periodic or adiabatic boundary conditions on the sides of the domain.

In terms of length scales, there are two important small-scale quantities. One is the thermal boundary layer δ_θ, which is the thin region adjacent to the walls where most of the temperature gradient of the fluid is concentrated. In this layer, temperature changes significantly, while beyond it, the fluid temperature becomes comparatively uniform. Understanding how δ_θ scales with varying Rayleigh and Nusselt numbers gives insight into the dynamics of convective heat transfer. The thermal boundary layer thickness can be estimated by identifying the height at which the root mean square (r.m.s.) of the temperature deviates from linear behaviour near the plates.

The other length scale is the Kolmogorov scale $\eta_K(\boldsymbol{x}, t)$, which represents the size at which viscous forces damp out the smallest turbulent eddies and is given by (Scheel, Emran & Schumacher 2013)

$$\eta_K(\boldsymbol{x}, t) = \left(\frac{\nu^3}{\varepsilon(\boldsymbol{x}, t)} \right)^{1/4}, \tag{2.6}$$

where $\epsilon(\boldsymbol{x}, t)$ is the kinetic energy dissipation rate that is defined as

$$\epsilon(\boldsymbol{x}, t) = \nu \tfrac{1}{2} (\nabla \boldsymbol{v} + \nabla \boldsymbol{v}^T)^2. \tag{2.7}$$

Due to the symmetries of the problem, η_K should be homogeneous in time and the horizontal direction but not in the vertical direction. Further details on how this behaves are given below when the datasets used are described.

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

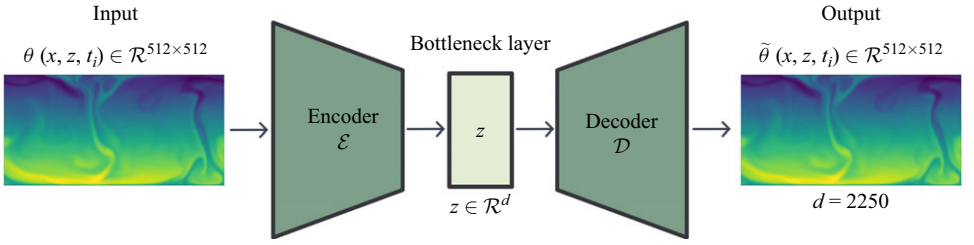


Figure 1. Two-dimensional convolutional neural network-based autoencoder. The central layer z in the middle is the bottleneck layer or latent vector. The image on the left-hand side represents an example of the input to the network for $Ra = 10^8$, corresponding to the temperature fields of a Rayleigh–Bénard numerical simulation. The image on the right is the output of the network for the same input, with a latent space dimension of $d = 2250$.

Finally, the characteristic time scale of the system is the large eddy turnover time, denoted by τ. This is calculated as the time required for a large-scale eddy to complete one full rotation over the height of the domain $\tau = 2h/w$, with w the r.m.s. of the vertical velocity component, $w \equiv v_{z,rms}$.

2.2. *Fixed- d autoencoders*

Autoencoders are a type of dimensionality reduction algorithm that seeks to construct an encoding function $\mathcal{E}$ that can map an input field $u \in \mathbb{R}^{N_x \times N_z}$ into a smaller-sized latent vector z and a decoding function $\mathcal{D}$ that can perform the inverse operation. This is done by approximating $\mathcal{E}$ and $\mathcal{D}$ with neural networks and training the resulting model $\mathcal{F}(u) = \mathcal{D}(\mathcal{E}(u))$ to approximate the identity operation $u = \mathcal{F}(u)$, where the latent vector $z = \mathcal{E}(u)$ acts as a bottleneck layer. The training process involves minimizing a prescribed loss function $\mathcal{L}(u, \mathcal{F}(u))$ with respect to the parameters of the neural networks. The most common choice of loss function (and the one used in the majority of this work) is the L_2 norm

$$\mathcal{L}(u, \tilde{u}) = \frac{1}{B}\sum_{b=1}^{B} L_2(u^{(b)} - \tilde{u}^{(b)}) = \frac{1}{B}\sum_{b=1}^{B}\sum_{i=1}^{N_x}\sum_{j=1}^{N_z}\left(u_{ij}^{(b)} - \tilde{u}_{ij}^{(b)}\right)^2, \tag{2.8}$$

where $\tilde{u}^{(b)} = \mathcal{F}(u^{(b)})$, and $u_{ij}^{(b)}$ and $\tilde{u}_{ij}^{(b)}$ are the individual grid elements of each field and B denotes the batch size, representing the number of training samples processed simultaneously during each iteration of the training.

A diagram of the autoencoders used is shown in figure 1. Our input data are snapshots of the rescaled temperature field $\theta(x, z, t_i)$ and the output is denoted as $\tilde{\theta}$. For simplicity, we also define $\Delta\theta = \theta - \tilde{\theta}$. We used convolutional neural networks for both the encoder and decoder as is standard in image-processing tasks. A schematic of the encoder architecture is shown in figure 2. The input fields of size $\mathcal{R}^{512\times512}$ undergo transformations through four convolutional layers, each employing a 5×5 kernel (highlighted in green in the figure). The configuration of each subsequent layer is characterized by its shape, which includes the number of channels and the new dimension size. The reduction in size at each layer is achieved through the application of specific strides and padding. After the final convolutional layer, the output is flattened to maintain its structure , which is a vector of dimension d', followed by the introduction of a dense layer with a smaller dimension d. This additional layer serves to form the compact encoded representation of the input, known as the bottleneck layer. The different hyperparameters used are listed in table 1.

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

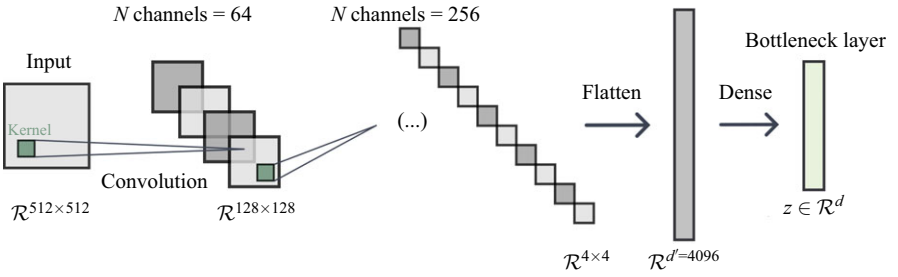


Figure 2. Schematic of the convolutional encoder $\mathcal{E}$. The first layers are convolutional while the last layer after the flattening operation is a fully connected layer. Details are listed in table 1.

Note that under this set-up the dimension of the latent vector d is a hyperparameter that has to be fixed prior to training. The primary goal of this work is to evaluate the performance of the autoencoder as we vary d in order to assess what is the minimum d needed by the autoencoder to properly represent the data, in this case the temperature field. We refer to this architecture as FdAE, variations are discussed below. Details about the training protocol and datasets used are given below. Note that, for the value of d', we estimated the minimum number of degrees of freedom necessary to represent the flow by determining the number of Fourier and Chebyshev modes needed to reconstruct the relevant small-scale dynamics. In the periodic x direction, we need at least $2k_\eta$ modes. For the non-periodic z direction, we can calculate the number of interpolation Chebyshev points needed to resolve the boundary layer. This estimation indicates that at $Ra \approx 10^8$ approximately 4096 modes are required (or degrees of freedom), thus determining the largest size (biggest bottleneck layer) to which we compressed our data using the networks.

2.3. *Variable- d autoencoders*

In the FdAE outlined above, the dimension of the latent space d has to be chosen prior to training and a new neural network has to be trained for each latent space d. We now discuss two regularized variations where d can be adjusted after training. In both variations, the networks are trained using an over-parametrized bottleneck layer of size $d' \gg d$ of which only d parameters are used for inference. We then always use d to refer to the effective dimension of the encoding.

2.3.1. *Sparsity-inducing autoencoders*

The first approach involves applying a L_1 regularization term, acting over $\boldsymbol{z}$, to the loss function (2.8), resulting in

$$\mathcal{L}_S(u, \tilde{u}, \boldsymbol{z}) = \mathcal{L} + \frac{1}{B}\sum_{b=1}^{B} \lambda L_1(\boldsymbol{z}^{(b)}), \tag{2.9}$$

where $\mathcal{L}$ is defined in (2.8) and B denotes the batch size. The regularization term is defined as

$$L_1(\boldsymbol{z}) = \frac{1}{d'}\sum_{j=1}^{d'} \left|z_j\right|, \tag{2.10}$$

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

Input shape	512×512
Encoder structure	$64 : 64 : 128 : 256$
Decoder structure	$256 : 128 : 64 : 64$
Kernel size	5×5
Strides	$4 : 4 : 4 : 2$
d'	4096
Activation function	ReLU
Last activation function	Tanh
L_1 regularization	$\lambda \in [10^{-3}, 10^{-1}]$
Implicit regularization	$n \in [1, 4]$

Table 1. The CNN autoencoder parameters. The encoder and decoder structure specify the number of filters per convolutional layer.

λ is an extra hyperparameter that controls the intensity of the regularization, with $\lambda = 0$ recovering the previous case.

By introducing this regularization, the model is encouraged to maintain only the most essential connections and promote sparsity in the middle layer. This approach was explored in Sondak & Protopapas (2021), where the authors employ an autoencoder with latent space penalization. We refer to this variation as a SIAE.

As stated above, a SIAE is trained using a bottleneck layer of size d'. After training, the respective latent representation $\boldsymbol{z}'$ of each element of the training dataset is calculated, forming a matrix Z', from which then the interquartile range (IQR) of each dimension is obtained. Finally, a sparse version of the latent vector $\boldsymbol{z}'$ is constructed where the $d' - d$ dimensions with the lowest IQR values are set to zero (and, thus, only d non-zero dimensions remain). This whole operation can be viewed as replacing the encoder $\mathcal{E}$ by

$$\mathcal{E}_S(\boldsymbol{x}) = \Theta_d(\mathcal{E}(\boldsymbol{x})), \tag{2.11}$$

where Θ_d is the operator that sets the least import dimensions to zero, during inference, meaning, too, that the latent vector takes the form $\boldsymbol{z} = \Theta_d(\boldsymbol{z}')$. The decoder is unchanged in this version. The structure of $\mathcal{E}$ and $\mathcal{D}$ themselves is the same as in the FdAE, except for the fact that the bottleneck layer has size d', of course, this is true in the next version as well. The values of d' and λ are presented in table 1.

2.3.2. *Implicit rank minimizing autoencoders*

The second approach is known as implicit regularization (IRMAE, Zeng & Graham 2023). This method uses the fact that adding depth to a matrix factorization enhances an implicit tendency towards low-rank solutions to construct low-rank versions of the latent representations (Jing *et al.* 2020). It works by adding n linear layers, $\mathcal{W} = \mathcal{W}_1, \mathcal{W}_2, \ldots, \mathcal{W}_n$, between the encoder and the bottleneck layer.

Similar to the SIAE, an IRMAE is trained with bottleneck layers of size d'. After training, the covariance matrix of the encodings Z' is formed, but now instead of calculating the IQR values, a singular value decomposition, $Z' = U \Sigma U^*$, is performed. The final latent vector is obtained by projecting into the first d dimensions. The resulting encoder used during inference has the form

$$\mathcal{E}_I(\boldsymbol{x}) = \Theta_d(U^* \mathcal{W} \mathcal{E}(\boldsymbol{x})), \tag{2.12}$$

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

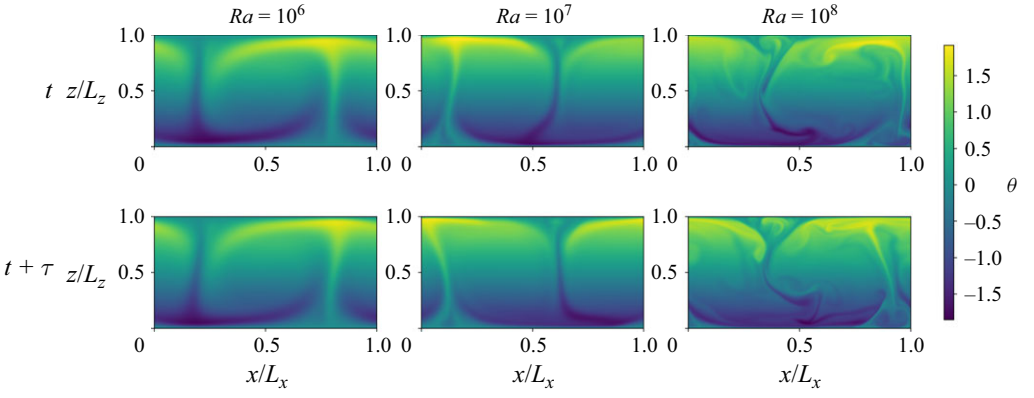


Figure 3. Visualizations of $\theta(x, z, t)$ and $\theta(x, z, t+\tau)$ for three different values of Ra.

where Θ_d truncates the singular values to those that are different from zero so that $U^*\boldsymbol{z}' \approx \Theta_d(U^*\boldsymbol{z}')$ holds. The decoder, which now needs to project back into Z', takes the form

$$\mathcal{D}_I(x) = \mathcal{D}(U\mathcal{E}_I(\boldsymbol{x})). \tag{2.13}$$

Contrary to Zeng & Graham (2023), we keep the loss function the same as in the FdAE (2.8) and do not add extra terms as we did not find that they changed the final results.

2.4. *Numerical simulations, datasets and training protocol*

Equations (2.1)–(2.3) where solved using SPECTER (Fontana *et al.* 2020), a highly parallel pseudo-spectral solver that utilizes a Fourier-continuation method to account for the non-periodic boundary conditions in the vertical direction. For implementation reasons, the code solves for the rescaled temperature θ, instead of the temperature T. So, as explained above, we present our results in terms of θ, this does not pose a problem for our analysis as the temperature fluctuations and the rescaled temperature fluctuations only differ by the scaling factor. We solved the equations at seven different Rayleigh numbers, ranging from 10^6 to 10^8. In all cases, the Prandtl number was set to $Pr = 1$, and the domain sizes were $[L_x, L_z] = [2\pi, \pi]$, with $N_x = N_z = 512$ grid points for spatial resolution in each direction. This spatial domain ensured that the smallest scales were resolved for all values of Ra. The values of the viscosity used at each $Ra = \{10^6,\ 5\times10^6,\ 7.5\times10^6;\ 10^7,\ 1.75\times10^7,\ 2.5\times10^7,\ 5\times10^7,\ 10^8\}$ were $\nu = \{9.86\times10^{-3},\ 4.41\times10^{-3},\ 3.60\times10^{-3},\ 3.12\times10^{-3},\ 2.36\times10^{-3},\ 1.97\times10^{-3},\ 1.39\times10^{-3},\ 9.68\times10^{-4}\}$, respectively. Simulations were started from random initial conditions for the temperature field, allowed to reach a statistically stationary state and then run for over 700 turnover times. For each Ra value, we conducted two simulations with different initial conditions.

We now take a moment to describe three paradigmatic cases, $Ra = 10^6$, 10^7 and 10^8. Figure 3 shows snapshots of $\theta(x, y)$ at a time t and one turnover time later, at a time $t+\tau$. As the Rayleigh number increases, the flow becomes more complex with smaller and smaller structures popping up. New time scales appear in the flow too, as can be in seen in figure 4, where we show the temporal evolution of θ at one single point (x_p, z_p) throughout several turnover times (while the location x_p is indistinct due to the translational symmetry of the flow, the significance of z_p will be discussed shortly). The turnover time τ is defined for each Ra. Even at $Ra = 10^7$ the flow exhibits multi-time-scale behaviour and

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

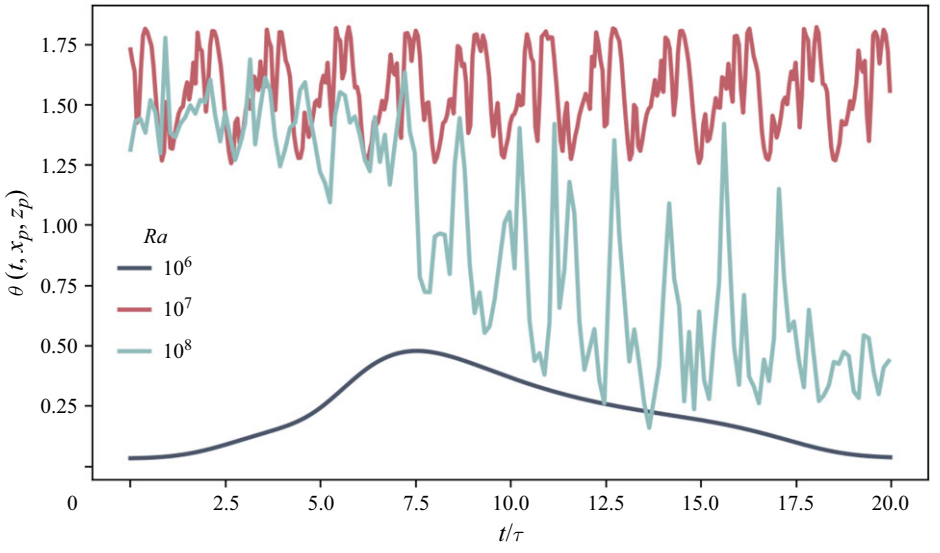


Figure 4. Temporal evolution at a fixed location (x_p, z_p) of the flow for three different Ra.

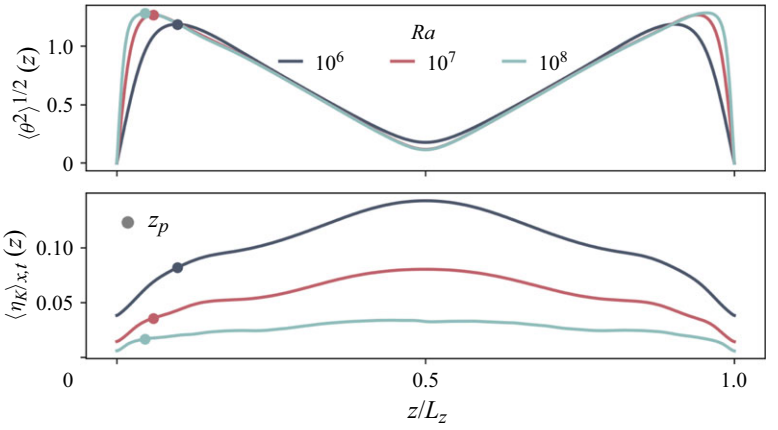


Figure 5. Relevant scales of the fluid for three different Ra. The top plot illustrates the profile of $\langle\theta^2\rangle^{1/2}_{x,t}$. The lower plot shows the Kolmogorov scale, $\langle\eta_K\rangle_{x,t}(z)$. For each Ra, z_p is marked.

intricate dynamics. The shift observed at $t = 7.5\tau$ in the $Ra = 10^8$ case corresponds to a horizontal displacement of the plume. Finally, we show profiles of the horizontally averaged and time-averaged r.m.s. fluctuations, $\langle\theta^2\rangle^{1/2}_{x,t}(z)$, and Kolmogorov lengths scale, $\langle\eta_K\rangle_{x,t}(z)$, in figure 5. As the Rayleigh number increases, the thermal boundary layer becomes steeper, as expected. As the dissipation is not uniform in the vertical direction, the Kolmogorov length scale increases when approaching the middle of the domain. We are interested in checking how well the autoencoders represent each scale of the flow, for this matter we define $z_p(Ra)$ as the the height where the temperature perturbation fluctuations are highest. Below z_p the dynamics is dominated by the boundary layer, while above the Kolmogorov length scale decreases when approaching the centre of the domain.

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

Ra	10^6	10^7	10^8
$\eta_K(z_p)$	0.06	0.03	0.01
Re	300	840	3300
Nu	7.7	14	25
δ_θ	0.07	0.04	0.02

Table 2. Parameters used for the reference Rayleigh–Bénard flows, all in non-dimensional simulation units. Here $Re = u_{rms}h/\nu$ denotes the Reynolds number, where u_{rms} is the r.m.s. velocity.

The corresponding z_p are marked in figure 5. As we will explain below, corroborating if an autoencoder is properly representing this scale will be the benchmark used to test its capabilities and choose the minimal number of dimensions d^* needed for the latent space dimension. As a summary, we list the values of $\eta_K(z_p)$, Re, Nu and δ_θ for the three Rayleigh numbers cases showcased in table 2.

As input for the networks, we utilized the two-dimensional θ fields from the numerical simulations described above, selecting specific snapshots for each dataset. We excluded the transient regime of the flow, as we are interested in the statistically steady states. The cutoff point for the transient regime was defined as the point when the volume-averaged kinetic energy began to stabilize, oscillating around a constant value. To address the continuous spatial symmetry in the horizontal direction, we augmented the data by applying 10 random shifts over the x axis at each time step. This augmentation increased the number of temperature fields under consideration from 7000 to 70 000, spanning over 700 turnover times. Note that it would also have been possible to deal with this symmetry by performing phase shifts, as described in Linot & Graham (2020) and formulated in Budanur *et al.* (2015), instead of data augmentation, we chose to explicitly retain the symmetry as we are interested in exploiting it in future analyses (Page *et al.* 2021, 2023). The datasets were normalized with respect to the global maximum and minimum, in order to feed $\theta/\theta_0 \in [-1, 1]$ to the neural networks. Additionally, a separate, longer dataset was created for testing to enable the integration of phenomena over extended periods, facilitating a more comprehensive evaluation of the performance of the model over longer time scales for certain physical metrics, such as the estimation of the Nusselt number.

During the training process, we employed the Adam optimizer with its default configurations to minimize the L_2 norm (2.8), which was also monitored on the validation set. Each network was trained over 500 epochs, by which point all networks had already stabilized. We monitored the validation metrics, and the batch size was set to 32. To ensure consistency, three different models were trained for each variation of the latent space dimension d. Each model was trained using different snapshots from the same extensive simulation, with both the training and test datasets maintained at the same size. Specifically, for each model, we performed three independent runs: (a) using a random 80–20 train-test split, (b) reinitializing the network with different learning rates and initialization seeds but keeping the same data split, and (c) using the same initialization but applying a different random 80–20 train-test split. In this last case, we also introduced new random shifts to the data. The results shown in the next section reflect the averaged values from these three variations.

3. Results

All results shown here are calculated over elements of the test datasets and averaged over them when appropriate.

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

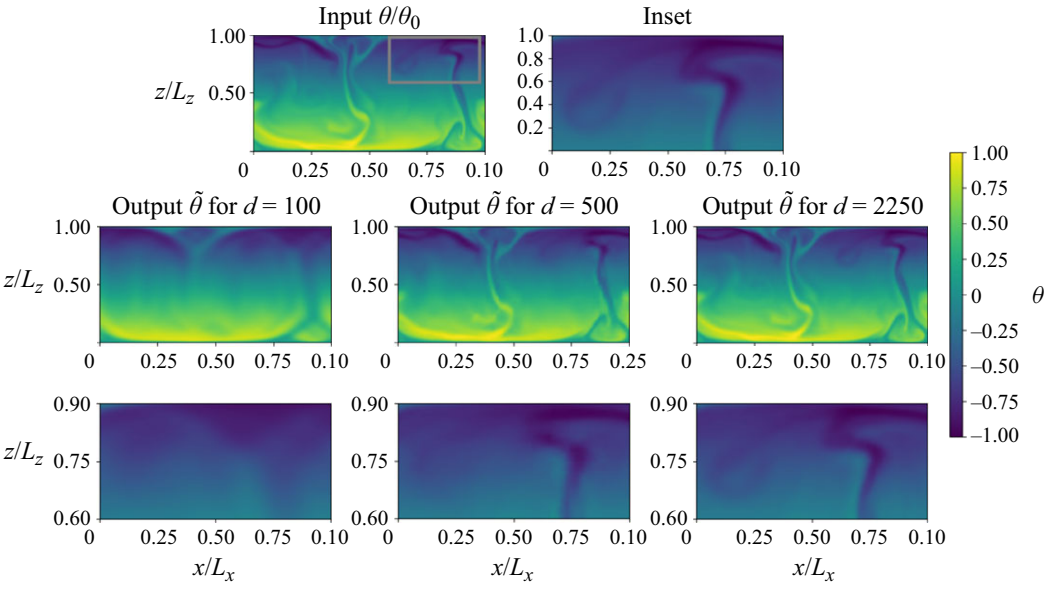


Figure 6. The output fields $\tilde{\theta}$ of the FdAE for a snapshot of the temperature field, corresponding to latent space dimensions $d = 100$, 500 and 2250, with respective MSEs of 1.6×10^{-3}, 3.1×10^{-4} and 4.0×10^{-5}, are shown. For comparison, the normalized ground truth field θ/θ_0, used as input, is also displayed. Additionally, an inset highlights a region where small-scale structures are present.

3.1. *Fixed- d autoencoders*

We focus our analysis on the performance of the FdAE on the $Ra = 10^8$ case, as this is the highest Rayleigh number used in our study, and thus, the most complex case. We reference results at other Rayleigh numbers when appropriate. At this particular value of Ra the flow is characterized by a rising hot plume and falling cold plume, which generate two large counter-rotating vortices that are accompanied by a rich variety of small-scale structures, the FdAE has to be able to capture the whole range of scales. In figure 6 we present a snapshot of the original temperature field θ (chosen out of the test dataset) together with its respective output $\tilde{\theta}$ obtained for selected values of latent space dimension d, along with zoomed-in views of regions with small-scale structures. These values (100, 500 and 2250) are noteworthy as they represent critical thresholds in performance and mark the transition points in the ability of the model to go from just reconstructing the large convective rolls to reproducing the small-scale details.

When $d = 100$, the FdAE mainly reconstructs the broader flow patterns and larger convective rolls. Increasing the latent space dimension to $d = 500$ enhances the performance of the autoencoder, which can now capture mid-range structures. At this value of d, the model begins to accurately reproduce physically relevant quantities, such as the boundary layer thickness δ_θ. In figure 7 we show the profile of θ_{rms} along with its reconstructions for $d = 100$ and $d = 500$. The inset highlights the near-wall region where the boundary layer thickness δ_θ is defined as the height z at which the θ_{rms} profile departs from a linear profile. While the $d = 100$ reconstruction does not accurately capture θ_{rms} in the bulk, the $d = 500$ model successfully reproduces the boundary layer structure. Finally, at $d = 2250$, the autoencoder is visually able to express the finer details of the

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

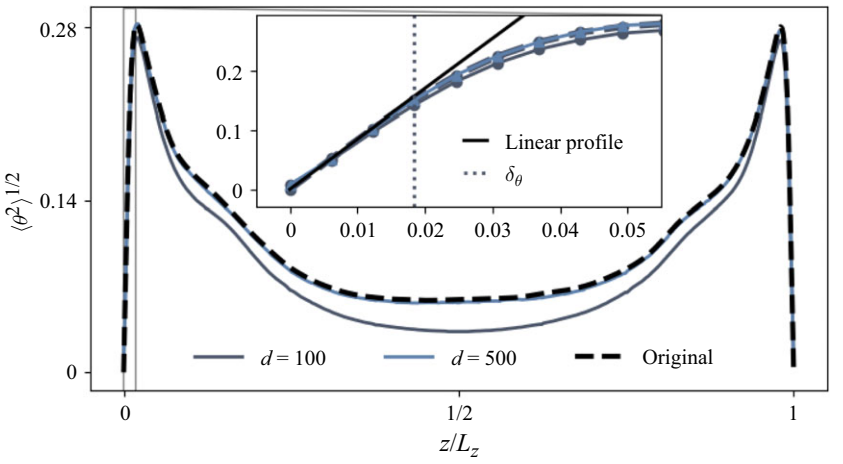


Figure 7. Profile of the original θ_{rms} along with reconstructed profiles for FdAE with $d = 100$ and $d = 500$. The inset zooms in on the near-wall region, showing the boundary layer thickness δ_θ, defined as the height z where the θ_{rms} profile transitions away from a linear profile. While the reconstruction with $d = 100$ does not capture θ_{rms} accurately in the bulk, the reconstruction with $d = 500$ successfully reproduces the boundary layer behaviour.

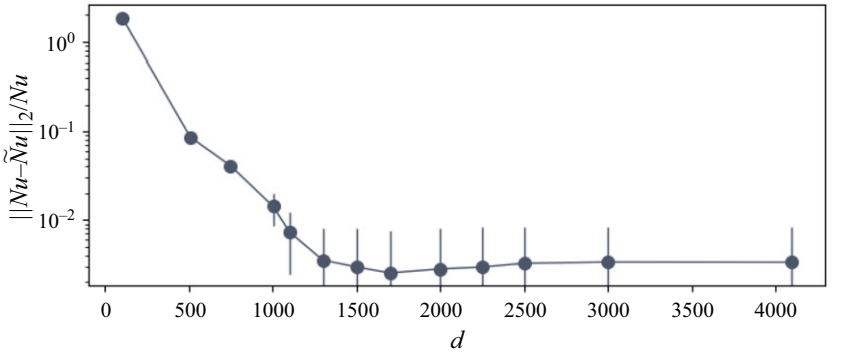


Figure 8. Difference between the Nusselt number Nu and the reconstructed Nusselt number $\tilde{Nu}$ with FdAE as a function of the latent dimension d. The error stabilizes at $d = 1500$, indicating that the model reliably captures Nu at this dimension. For reference, the true value of Nu is provided in table 1.

flow. To further quantify this, we examine the accuracy of Nu as a function of latent dimension d. Figure 8 shows the difference between the true Nusselt number Nu and the reconstructed Nusselt number $\widetilde{Nu}$ across varying d. This difference stabilizes around $d = 1500$, indicating that the model reliably captures Nu at this dimension. These figures collectively serve as a visual testament to the power of the FdAE in identifying and reconstructing features across various scales, highlighting the importance of latent space dimensionality in the performance of the model. One of the main goals of this paper is to introduce a clear metric to determine the specific value of d at which all relevant scales are accurately reconstructed, as discussed in detail below.

Figure 9 shows the mean squared errors (MSEs) between input and output, MSE($\Delta\theta$), as a function of latent space dimension, d. Additionally, the plot includes comparative

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

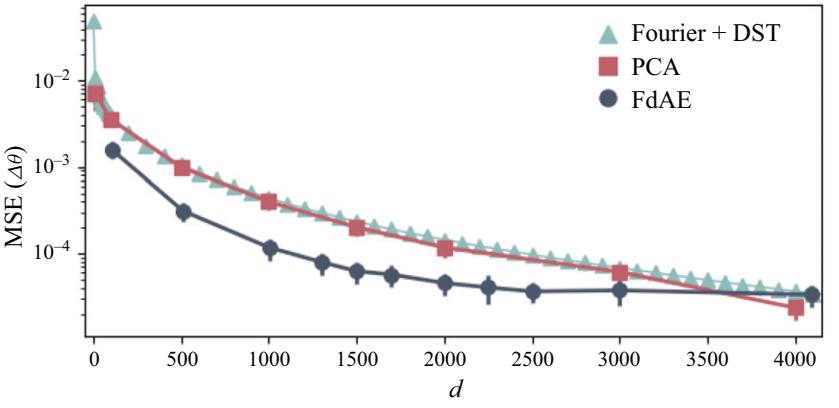


Figure 9. Comparison of the MSE of the temperature field (θ) reconstructions against the latent space dimension d for the FdAE (represented by circles), PCA (illustrated with squares) and Fourier-DST decomposition (denoted by triangles) using the same dataset. For the autoencoder, each data point corresponds to the average of three runs with minor variations.

lines representing the results of PCA and a decomposition combining Fourier and the discrete sine transform (DST) in the z (non-periodic) direction for different latent space dimensions, i.e. the number of modes used in each case. Such comparisons are instrumental in benchmarking the performance of the FdAE against these established linear techniques. To ensure a fair comparison, when doing PCA, we projected the training dataset and then evaluated the test data using the modes obtained from the initial decomposition. It is important to note that due to the large size of the training dataset matrix, which stacks all the flattened images, performing PCA directly is too computationally intensive. Consequently, we resorted to using incremental PCA from Scikit-learn (Pedregosa *et al.* 2011), which is a suitable approximation method for large datasets that computes PCA over small batches.

In neither case does the MSE exhibit a sharp drop-off that would indicate the minimum number of dimensions needed to accurately represent the flow. This is to be expected in systems with this degree of scale separation, for example, when studying the Kuramoto–Sivashinsky equation, Linot & Graham (2022) found sharp decreases in the MSE at the d coinciding with that estimated for the inertial manifold only when using small domain sizes, when they increased the domain the drop-off became noticeably smoother. Similar results were observed in studies of turbulent pipe flows (Constante-Amores *et al.* 2024). This effect is only greater in our case. The FdAE outperforms the linear methods in terms of the MSE within the test dataset, demonstrating its capacity to capture the nonlinearity attributes present in the dataset. Furthermore, it is important to acknowledge that the MSE might not be the most fitting metric to gauge the fidelity of reconstructions, particularly for smaller scales, as it predominantly captures the high energy modes, which are effectively resolved even at lower latent space dimensions as demonstrated in figure 6.

To quantify the ability of the FdAE to represent different scales in the flow, we compute the energy spectrum of the difference between input and output:

$$E_{\Delta\theta}\left(k_x\right) = \left\langle \left| \Delta\hat{\theta}\left(k_x, z_p, t\right) \right|^2 \right\rangle_t . \tag{3.1}$$

Here $\Delta\hat{\theta}(k_x, z_p, t)$ are the Fourier coefficients of $\Delta\theta$ in the horizontal direction at height z_p and $\langle , \rangle_t$ is a time average. The resulting spectra, calculated for $d = 100$, 500, 2250

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

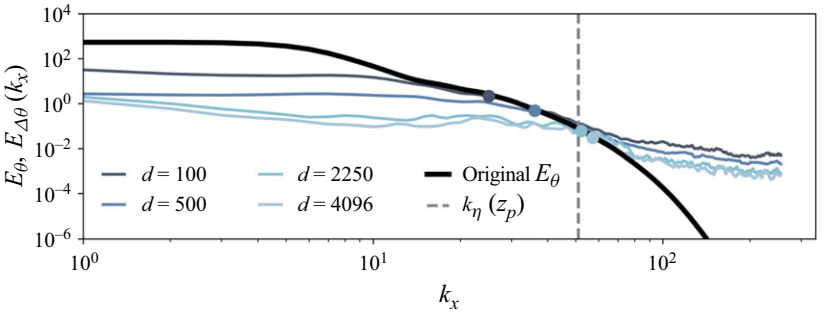


Figure 10. Error spectra $E_{\Delta\theta}$ for various reconstructions, alongside the reference energy spectra E_θ for comparison. The spectra were calculated at z_p. The dotted vertical line indicates the wavenumber corresponding to the Kolmogorov scale k_η also at z_p, and the dots represent the maximum resolved k_x for each d.

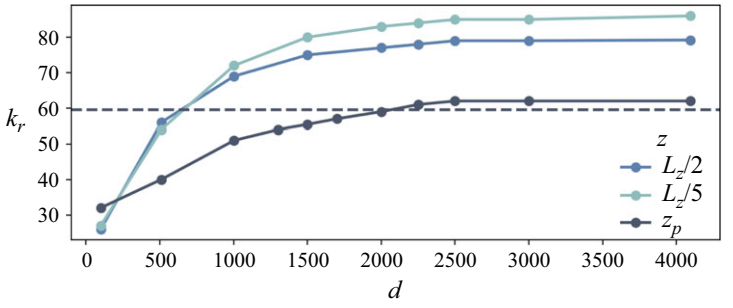


Figure 11. Maximum k_x^* resolved by each FdAE for different heights. The horizontal line shows the Kolmogorov scale k_η of the flow at z_p. At $d^* = 2250$ the network reconstructs scales up to k_η for z_p.

and 4096, are shown in figure 10, alongside the spectrum of the original test dataset as reference. For $d = 100$, the error spectrum has large values (compared with the original spectrum) across almost all scales, indicating the inability of the FdAE to accurately represent the flow. As the latent space dimension increases, the FdAE progressively improves at capturing features corresponding to smaller wavenumbers k_x. At $d = 2250$, the FdAE can faithfully represent scales up to the Kolmogorov scale $k_\eta(z_p)$, therefore, completely describing the flow, as expected from the results above. The significance of z_p and the justification for choosing it as the relevant height is given below.

By inspecting figure 10, we can define the resolved wavenumber k_r for each d as the first wavenumber k_x at which the corresponding $E_{\Delta\theta}$ intersects with E_θ (the spectrum of the input). This intersection represents the smallest scale at which the output of the networks fails to accurately reproduce the input, effectively indicating the maximum k_x that is resolved for each d. This is shown in figure 11, where we observe that the FdAE networks start to reproduce scales up to $k_\eta(z_p)$ for $d = 2250$. Note that in this figure we also repeat the analysis at heights closer to the centre of the channel. At these higher heights not only is the Kolmogorov length scale larger, as seen in figure 5, but the autoencoder is also able to resolve smaller scales at lower d. We recall too that at lower heights the dynamics is dominated by the boundary layer that is also resolved at lower values of d, as shown in figure 7. Therefore, when the autoencoder is able to resolve $\eta_K(z_p)$, it is also able to resolve the smallest scale at every height. We then define the minimum dimension of the

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

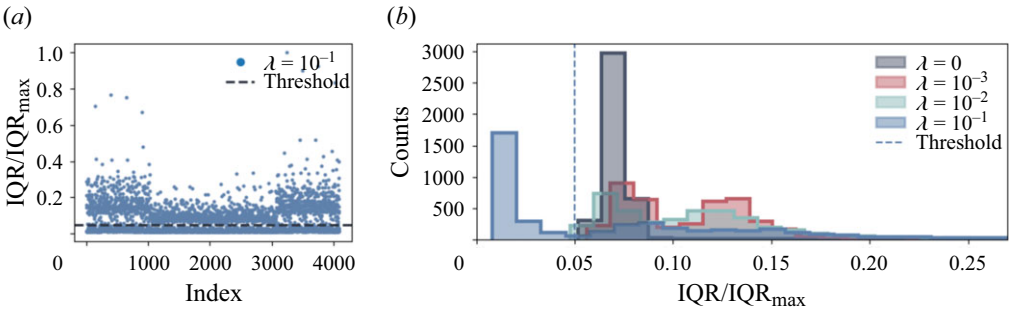


Figure 12. Distribution of the normalized IQR for the SIAE with different regularization values λ, where the normalization is done using the maximum IQR value. (*a*) The IQR values for the network with a regularization strength of $\lambda = 10^{-1}$, with the line marking a threshold that differentiates the two observed distributions. (*b*) The IQR distributions for various λ values, including a comparison with the F*d*AE ($\lambda = 0$, shown in darker shading). The line in this figure represents the same threshold as identified in (*a*).

latent space needed by the autoencoder to properly resolve all scales of the temperature field, d^*, as that for which $k_r \approx k_\eta$ at z_p. This definition acknowledges the multiscale nature of the flow and does not rely on finding a sharp drop-off in a given metric, a task that is not always possible. Note that if the autoencoder was handling other physical fields besides the temperature, this criteria would need to be expanded such that every scale of every field is accounted for. Therefore, at $Ra = 10^8$ we obtain $d^* = 2250$. We extend this analysis to other Rayleigh numbers below.

3.2. *Sparsity-inducing autoencoders*

We now turn our attention to the SIAE, while focusing again on the $Ra = 10^8$ case. We repeat the same experiments as above using a fixed bottleneck size of $d' = 4096$ and then proceed to analyse the structure of the latent vector. To identify which dimensions of Z' are relevant, we process each snapshot of the test dataset through the trained network, extract z' for each instance, and then calculate the IQR of each dimension of the vector as a measure of their variability. A large IQR indicates significant activity in that dimension during the reconstruction of the test dataset.

We tested different values of the regularization parameter λ. This led to varying distributions of the IQR of the bottleneck layer activations as seen in figure 12(*b*). We employed two criteria for selecting the regularization parameter. First, we evaluated the MSE of the reconstructions for different values of λ. After this, we analysed the IQR distributions produced by regularizations that had comparable MSE performance. We considered a good value for λ to be one whose bimodal histogram had a left distribution close to zero. This indicates that the neurons in the bottleneck layer corresponding to that distribution were not highly activated. As λ is increased, thereby enhancing the regularization strength, the separation between the two groups becomes more pronounced and the left tail of the distribution gets closer to zero IQR, at the expense of reconstruction accuracy as indicated by the MSE.

Figure 12(*a*) displays the normalized IQR (normalized by the largest IQR, corresponding to the most active dimension) across each latent space vector index using the chosen regularization strength $\lambda = 10^{-1}$. While models such as the Kuramoto–Sivashinsky or damped Korteweg–de Vries, which are characterized by a finite-dimensional manifold, exhibit a distinct separation in the IQR distribution (Sondak & Protopapas 2021), our high-dimensional Rayleigh–Bénard system does not demonstrate

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

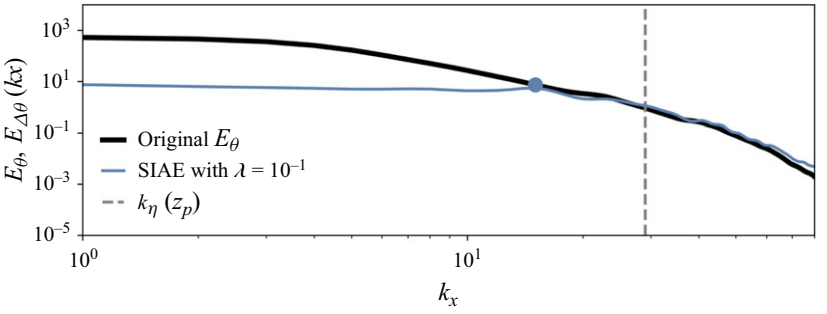


Figure 13. Error spectra $E_{\Delta\theta}$ for SIAEs with $\lambda = 10^{-1}$, along with the reference energy spectra E_θ for comparison. The spectra were calculated at z_p by taking the values from the latent space that corresponded to the right distribution of the histogram shown in figure 12(*b*) that has 2010 dimensions. The dotted vertical line indicates the wavenumber corresponding to the Kolmogorov scale k_η at z_p, and the dot represents the maximum resolved k_x.

such a clear demarcation at plain sight. Nevertheless, the IQR distribution histograms shown in figure 12(*b*) indicate two distinct groups of IQR values.

Regardless of the λ value for this range, the number of dimensions classified as more relevant (the number of indices whose normalized IQR falls into the second distribution on the right-hand side of the histogram) is around 2000, these identified dimensions are capable of reconstructing the flow with an MSE comparable to that achieved when using the entire bottleneck layer for decoding.

The SIAE does not provide an ordered set of relevant dimensions in the latent space. Instead, we selected all the values corresponding to the appropriate distribution of IQR (representing the significantly activated latent space dimensions). We then applied the same procedure used for estimating k_r and d^* as described earlier for FdAE, as shown in figure 13 for the case with $\lambda = 10^{-1}$. Unlike FdAE, SIAE does not reconstruct scales up to k_η for every height z. This was verified even for λ values that performed worse in terms of MSE. For small λ values, the criteria is met, but the IQR distribution does not differ significantly from that observed with $\lambda = 0$, resulting in a d^* estimate equal to the total size of the latent space, with no regularization.

3.3. *Implicit regularization*

Finally, we analyse the performance of the IRMAE. Figure 14 shows the singular value spectra of the latent space covariances of Z' for varying numbers of intermediate linear layers n. For comparison, we present the singular values obtained from PCA decomposition applied to the training data and then projected onto the test data. Additionally, we examine the spectra from the FdAE with $d = 4096$, indicative of no implicit regularization, to highlight the distinctions between these models in terms of regularization effectiveness. The PCA singular values remain relatively constant beyond the initial 10 values. However, with an increase in n, there is a notable drop in the singular values of IRMAEs at certain indices. Specifically, the FdAE exhibits a decrease towards the end of the index range, implying the utilization of over 3000 dimensions in the bottleneck layer to represent the data. As n increases, the drop starts to occur at lower index values; with $n = 1$, the reduction in singular values is minimal and gradual. In contrast, for $n = 2$, the drop is more pronounced. At $n = 3$, the regularization becomes too intense, as evidenced by the utilization of 1000 dimensions, which fails to achieve an effective reconstruction (in terms of the MSE and visual quality) as observed with $n = 2$ or lower.

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

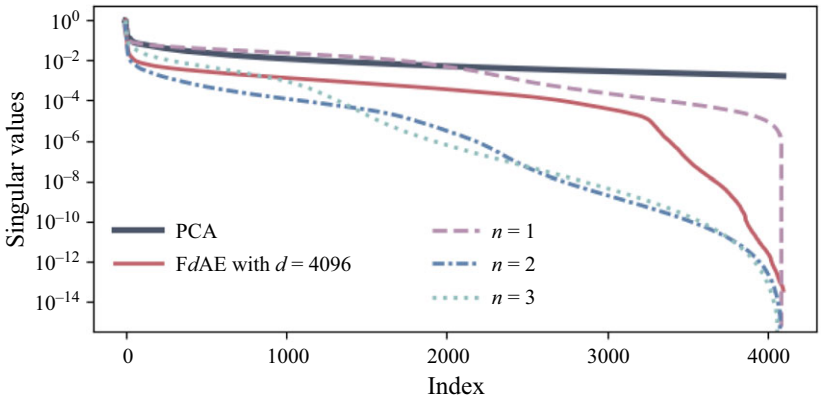


Figure 14. Singular values of the latent space. Each curve represents the singular values of the covariance matrix of the test dataset. Additionally, the thick black line illustrates the spectrum of singular values obtained by performing PCA on the training dataset and projecting the test dataset onto its modes. Also shown for comparison is FdAE with $d = 4096$. As we increase the number of linear layers n, we observe increased regularization.

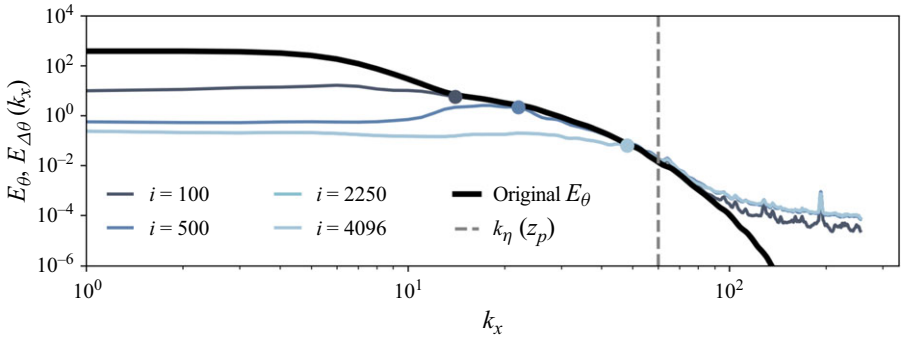


Figure 15. Error spectra $E_{\Delta\theta}$ for IRMAE with $n = 2$, along with the reference energy spectra E_θ for comparison. The spectra were calculated at z_p by taking the first i values from the latent space, based on the singular value spectra. Note that the lines for $d = 2250$ and $d = 4096$ overlap due to the similarity in their spectra, indicating that the IRMAE network does not improve significantly after some d. The dotted vertical line indicates the wavenumber corresponding to the Kolmogorov scale k_η at z_p, and the dots represent the maximum resolved k_x for each i.

For $n = 2$, the MSE is 3.5×10^{-5}, while for $n = 3$, it is an order of magnitude larger. When training networks whose dimensionality of the latent vector z' was below the cutoff value, the singular value spectra did not show significant reductions, thereby utilizing the full capacity of the layer.

As no sharp drop-off can be observed in the singular value spectra, as was observed in other, smaller, systems (Jing *et al.* 2020; Zeng & Graham 2023), this criterion cannot be used to estimate the minimum needed dimension of the latent space. Contrary to SIAEs, IRMAEs offer a natural ordering of the latent dimensions, and thus, the same procedure to estimate k_r and d^* used above for FdAE can be repeated here, as seen in figure 15 for the case with $n = 2$. Doing so shows that, under this set-up (which we tested thoroughly but does not exclude every possible architecture and hyperparameter choice), IRMAEs fail to accurately represent all scales, and thus, no estimate of d^* can be extracted. We examined

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

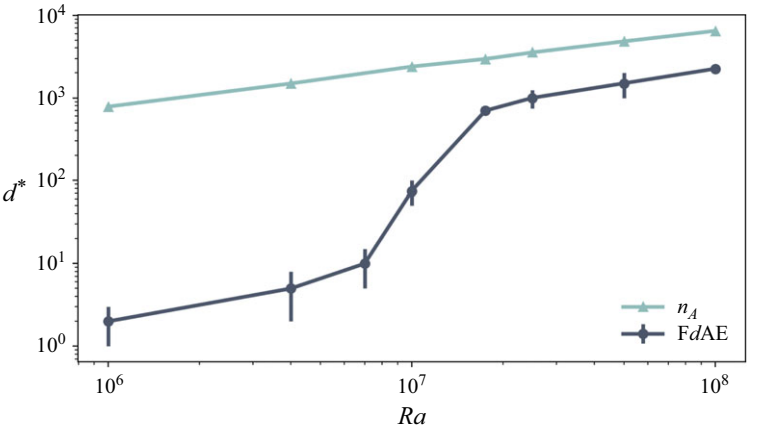


Figure 16. Minimum dimension needed to represent all relevant physical scales of the temperature field using FdAE, d^*, as a function of the Rayleigh number. An estimate of the number of degrees of freedom that scales as the inverse of $\delta_\theta\ \eta_K$ is also shown.

the dependence of IRMAEs on different values of N_{train} in Appendix B to verify that the inability to reproduce the Kolmogorov scale was not due to the size of the training dataset. It is important to note that we were able to get estimates of d^* at lower Rayleigh numbers, but they were consistently larger than those obtained via FdAE, more details on this below. Additionally, the MSE achieved by IRMAEs was slightly higher than that of FdAE, at 5.55×10^{-5} and 4.15×10^{-5} for $Ra = 10^8$, respectively.

3.4. *Minimal dimension at different Rayleigh numbers*

Finally, we are interested in estimating the minimal dimension needed to accurately represent every scale of the temperature field, d^*, at different Rayleigh numbers by extending he analysis for FdAE presented in § 2.2. This can be done without any modifications to the method outlined. As examples, the cases with $Ra = 10^6$ and 10^7 are shown in Appendix A. Figure 16 shows d^* as a function of the Rayleigh number. The uncertainty in the FdAE estimates is due to the discretization of the latent dimension sizes tested. For instance, for $Ra = 10^7$, the estimate for d^* was 75, based on tests with $d = 50$, $d = 75$ and $d = 100$. Since $d = 75$ was the first dimension to meet the criteria, the uncertainty is taken as 25, representing the interval between the tested dimensions. This error likely overestimates the true uncertainty. We also show n_A, an estimate of the degrees of freedom of the field calculated by taking the area of the domain and dividing it by the smallest physical area $\delta_\theta\ \eta_K$. As we could not get IRMAEs to work for the whole range of Rayleigh numbers studied, and as the values we were able to obtain were always considerably larger than those obtained via FdAE (for example, we obtained values of 50 and 450 for $Ra = 10^6$ and 10^7, respectively), we decided to not include them in the figure. The results obtained via FdAE show a clear transition in the dynamics of the flow between $Ra = 5 \times 10^6$ and 10^7. Below the transition d^* has a value of 5, around the transition it starts to grow quickly, and at around $Ra = 2.5 \times 10^7$, it slows down again and d^* appears to start to scale as n_A. Unsurprisingly, n_A does not exhibit different behaviours before and after the transition, as the scaling of δ_θ and η_K does not change (Siggia 1994; Grossmann & Lohse 2000; Scheel *et al.* 2013).

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

4. Conclusions

In this study we analysed the performance of different autoencoder architectures when trained to generate reduced-order representations of the temperature field of Rayleigh–Bénard flows as the flow transitions to turbulence. We quantified the error at each scale, and defined d^* as that needed to obtain a small error at every physically relevant scale. This criterion ultimately worked much better for multiscale flows than previously suggested ones. We used this methodology to calculate d^* at Rayleigh numbers ranging from 10^6 to 10^8. The results indicate a sharp increase in dimensionality as the flow transitions to turbulence, with d^* being equal to 2 at $Ra = 10^6$, and rapidly climbing several orders of magnitude after, reaching a value of 2250 at $Ra = 10^8$. Note that in this work we focused only on the temperature field. The values of d^* obtained serve then only as a proxy for the dimensionality of the whole system but are still indicative of its behaviour. Following past results in the synchronization properties of Rayleigh–Bénard flows (Agasthya *et al.* 2022), it can be expected that at low Ra the value of d^* obtained here for the temperature field is the same as that of the whole system, as the velocity and temperature fields are locked, while at higher Ra the coupling between the two fields decreases and dimensionality of the whole system is probably larger than just that of the temperature field. These statements will be checked empirically in future work. Having a parametrization of the submanifold where the dynamics live that captures the transition to turbulence opens an important door in studying the mechanisms of the transitions that will be explored in future work.

We also tested autoencoder architectures developed for fluid dynamics dimensionality reduction by Sondak & Protopapas (2021) and Zeng & Graham (2023), the SIAE and the IRMAE, respectively. Both architectures incorporate regularizations in the latent space to minimize its effective dimensionality. Despite their success with simpler systems, they failed to produce satisfactory results at every Rayleigh number studied here. At the Rayleigh numbers where they did work, they produced consistently higher results than FdAE. The quest for regularized architectures that can estimate d^* accurately in highly turbulent and multiscale flows remains open.

The dimensionality reduction achieved here gives an opportunity to employ neural ODEs for dynamic modelling. Neural ODEs could provide a powerful tool for modelling the time evolution of the flow's latent features, without needing the full computational complexity of the higher-dimensional space.

Acknowledgements. The authors would like to thank G. Torre, P. Mininni, M. Buzzicotti, L. Biferale and M. Sbragaglia for fruitful discussions. The authors would also like to thank the anonymous reviewers for their insightful comments.

Funding. This work was partially supported by the Google PhD Fellowship Program.

Declaration of interests. The authors report no conflict of interest.

Data availability statement. Autoencoder codes, as well as scripts and instructions to generate datasets, are available at Zenodo (10.5281/zenodo.14606152). Data are also available upon request.

Author contributions. P.C. conceived the idea and planned the work. M.V. generated data and trained the autoencoders. All authors contributed equally to analysing data and reaching conclusions, and in writing the paper.

Appendix A. Results for lower Rayleigh numbers ($Ra = 10^6$ and 10^7)

In the main text we focused on presenting detailed results for the highest and most complex Rayleigh number, $Ra = 10^8$, among the values considered. To provide a broader perspective on the system's behaviour at different Rayleigh numbers, this appendix

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

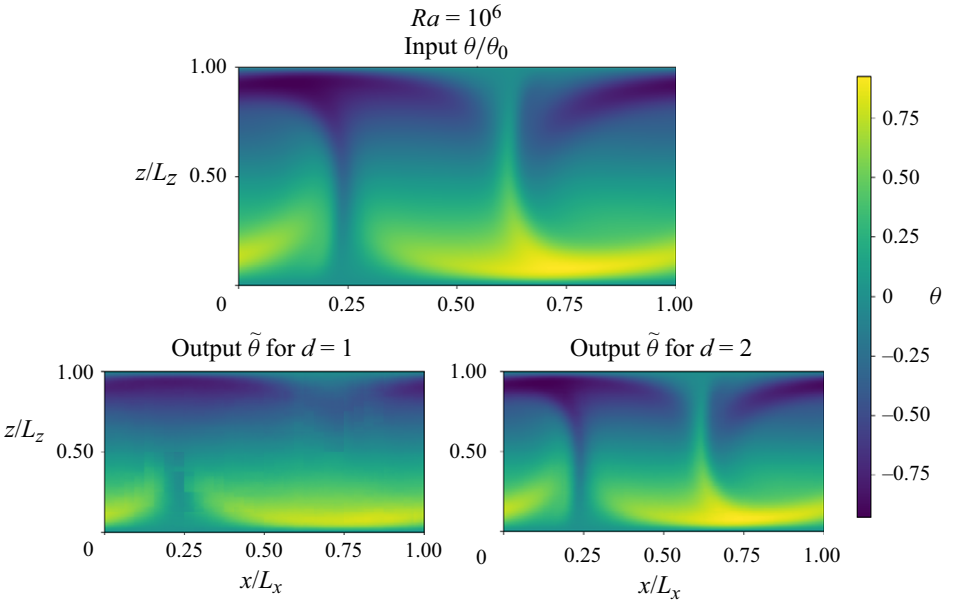


Figure 17. Output fields $\tilde{\theta}$ of the FdAE for $Ra = 10^6$, showing a snapshot of the output temperature field for latent space dimension $d = \{1, 2\}$. The value $d = 2$ corresponds to d^* for this Ra. The normalized ground truth field, θ/θ_0, used as input, is shown for comparison.

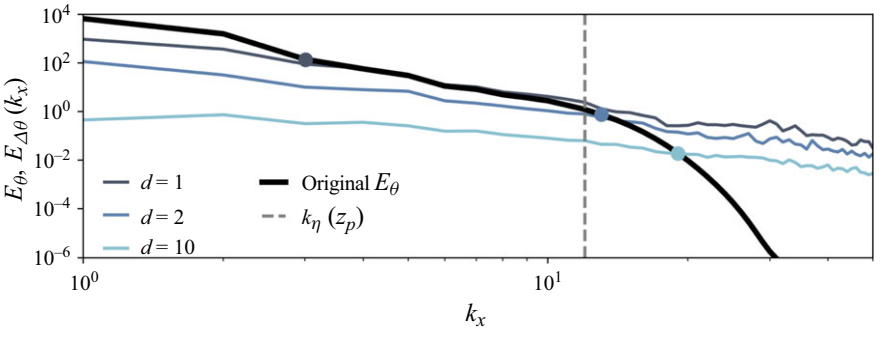


Figure 18. Error spectra $E_{\Delta\theta}$ for various reconstructions, alongside the reference energy spectra E_θ for comparison. The spectra were calculated at z_p. The dotted vertical line indicates the wavenumber corresponding to the Kolmogorov scale k_η also at z_p, and the dots represent the maximum resolved k_x for each d.

presents additional results for $Ra = 10^6$ and $Ra = 10^7$, where the minimum dimensions needed to represent the smallest scales of fluid dynamics are $d^* = 2$ and $d^* = 75$, respectively.

Here, we show visualizations of the FdAE output at different latent space dimensions, d, as seen in figure 6. The original normalized temperature field used as input for the networks is also presented for reference. For each Ra, we display two values of d: one below the corresponding d^* and the other at $d = d^*$, as shown in figures 17 and 19 for

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

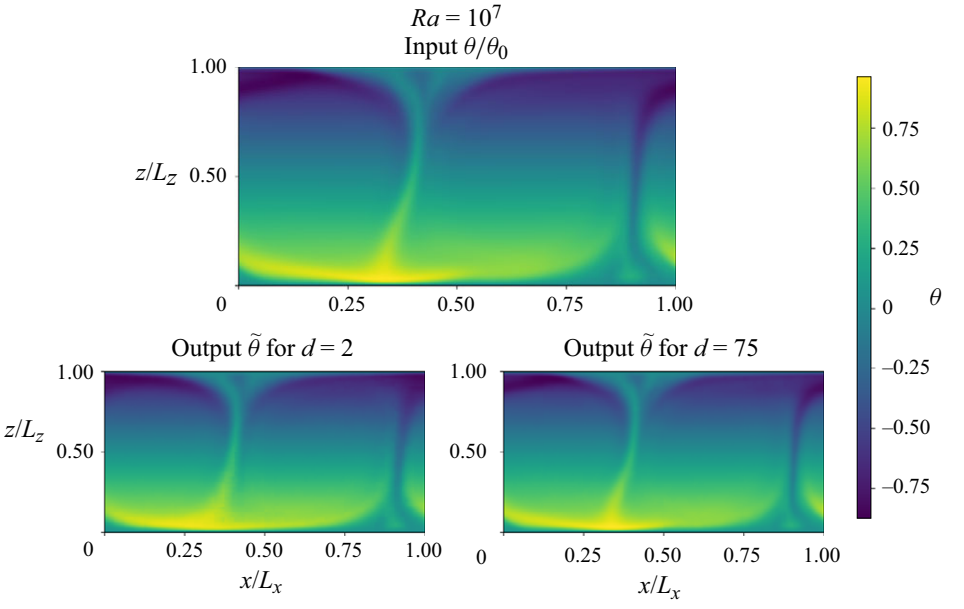


Figure 19. Output fields $\tilde{\theta}$ of the FdAE for $Ra = 10^7$, showing a snapshot of the output temperature field for latent space dimensions $d = \{10, 75\}$. The value $d = 75$ corresponds to d^* for this Ra. The normalized ground truth field, θ/θ_0, used as input, is shown for comparison.

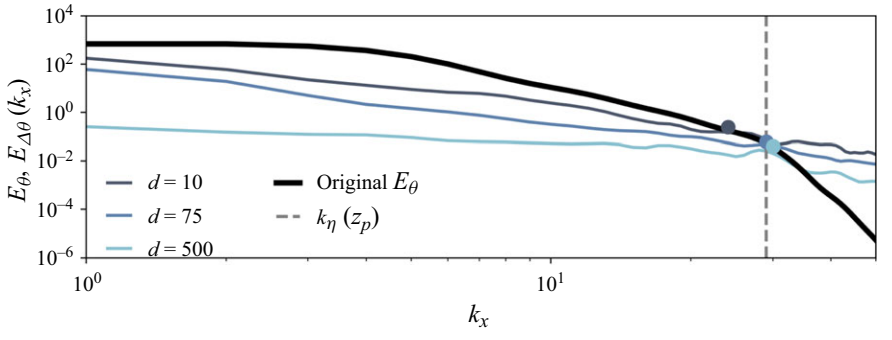


Figure 20. Error spectra $E_{\Delta\theta}$ for various reconstructions, alongside the reference energy spectra E_θ for comparison. The spectra were calculated at z_p. The dotted vertical line indicates the wavenumber corresponding to the Kolmogorov scale k_η also at z_p, and the dots represent the maximum resolved k_x for each d.

$Ra = 10^6$ and $Ra = 10^7$. We also present analogous figures to figure 10 for these two Ra values. For each case, we show the original energy spectrum E_θ and the difference spectra $E_{\Delta\theta}$ for different latent space dimensions, one at $d = d^*$ and one higher than d^* as shown in figures 18 and 20. Finally, we include the analogous to figure 11 for $Ra = 10^6$ and $Ra = 10^7$ in figure 21, which illustrates the maximum resolved wavenumber k_r achieved by each FdAE model across different latent space dimensions.

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

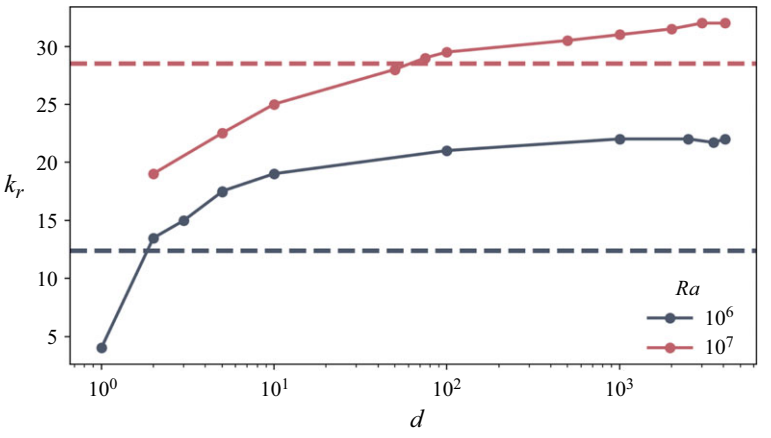


Figure 21. Maximum k_x^* resolved by each FdAE for $Ra = 10^6$ and $Ra = 10^7$, as seen in figure 11. The horizontal dashed lines show the Kolmogorov scale k_η of the flow at each z_p for both Rayleigh numbers.

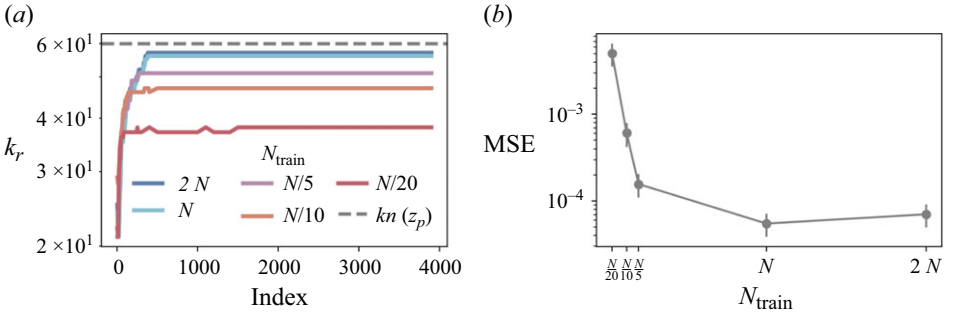


Figure 22. (*a*) Maximum wavenumber resolved by each IRMAE network for different numbers of training images, N_{train}, at $Ra = 10^8$. The horizontal line indicates the Kolmogorov scale k_η for this Rayleigh number. (*b*) The MSE of the validation dataset for the temperature field for each network.

Appendix B. Impact of training dataset size on IRMAE performance

At the highest Rayleigh number considered, $Ra = 10^8$, IRMAEs failed to accurately represent all scales of the flow, and thus, no estimate of d^* can be extracted. We conducted several experiments at the highest Rayleigh number considered, $Ra = 10^8$, to verify this. In these experiments, we varied the amount of data used for training, denoted by N_{train}. The performance of the trained autoencoders was evaluated using a consistent test set, matching the size of the test set used throughout the study. This procedure was applied to IRMAEs, leveraging the fact that we can train a single network for all d values.

As shown in figure 22, the performance of the autoencoders, measured in terms of the maximum resolved wavenumber k_r, improves with an increase in N_{train}. It is important to note that all the autoencoders exhibit similar performance for small wavenumbers k, except for the case with a low amount of training data ($N_{train} = 100$, augmented with 10 rolls). Indeed, increasing N_{train} results in the reconstruction of more k_x values, but even with a dataset twice as large, we were unable to resolve scales up to k_η, with only

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

marginal improvement in the maximum resolved k_x. We also evaluated the MSE across these experiments, observing minimal improvement as N_{train} increases.

References

Agasthya, L., Clark Di Leoni, P., & Biferale, L. 2022 Reconstructing Rayleigh–Bénard flows out of temperature-only measurements using nudging. *Phys. Fluids* **34** (1), 015128.

Brunton, S.L. & Kutz, J. 2019 *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press.

Budanur, N.B., Cvitanović, P., Davidchack, R.L. & Siminos, E. 2015 Reduction of SO(2) symmetry for spatially extended dynamical systems. *Phys. Rev. Lett.* **114** (8), 084102.

Castillo-Castellanos, A., Sergent, A., Podvin, B. & Rossi, M. 2019 Cessation and reversals of large-scale structures in square Rayleigh–Bénard cells. *J. Fluid Mech.* **877**, 922–954.

Chandler, G.J. & Kerswell, R.R. 2013 Invariant recurrent solutions embedded in a turbulent two-dimensional Kolmogorov flow. *J. Fluid Mech.* **722**, 554–595.

Chen, R.T., Rubanova, Y., Bettencourt, J. & Duvenaud, D. 2019 Neural ordinary differential equations. arXiv:1806.07366

Constante-Amores, C., Linot, A.J. & Graham, M.D. 2024 Dynamics of a data-driven low-dimensional model of turbulent minimal pipe flow. arXiv:2408.03135.

Crowley, C. J., Pughe-Sanford, J.L., Toler, W., Krygier, M.C., Grigoriev, R.O. & Schatz, M.F. 2022 Turbulence tracks recurrent solutions. *Proc. Natl Acad. Sci. USA* **119** (34), e2120665119.

De, J., Pérez, C.E. & Graham, M.D. 2023 Data-driven low-dimensional dynamic model of Kolmogorov flow. *Phys. Rev. Fluids* **8** (4), 044402.

Ding, X., Chate, H., Cvitanovic, P., Siminos, E. & Takeuchi, K. 2016 Estimating the dimension of an inertial manifold from unstable periodic orbits. *Phys. Rev. Lett.* **117**(2), 024101.

Eivazi, H., Le Clainche, S., Hoyas, S., & Vinuesa, R. 2022 Towards extraction of orthogonal and parsimonious non-linear modes from turbulent flows. *Expert Syst. Appl.* **202**, 117038.

Floryan, D. & Graham, M.D. 2022 Data-driven discovery of intrinsic dynamics. *Nat. Mach. Intell.* **4** (12), 1113–1120.

Foias, C., Sell, G.R. & Temam, R. 1988 Inertial manifolds for nonlinear evolutionary equations. *J. Differ. Equ.* **73** (2), 309–353.

Foldes, R., Camporeale, E. & Marino, R. 2024 Low-dimensional representation of intermittent geophysical turbulence with high-order statistics-informed neural networks (H-siNN). *Phys. Fluids* **36** (2), 026607.

Fontana, M., Bruno, O.P., Mininni, P.D. & Dmitruk, P. 2020 Fourier continuation method for incompressible fluids with boundaries. *Comput. Phys. Commun.* **256**, 107482.

Fukami, K., Nakamura, T. & Fukagata, K. 2020 Convolutional neural network based hierarchical autoencoder for nonlinear mode decomposition of fluid field data. *Phys. Fluids* **32** (9), 095110.

Goodfellow, I. J., Bengio, Y. & Courville, A. 2016 *Deep Learning*. MIT Press.

Grossmann, S. & Lohse, D. 2000 Scaling in thermal convection: a unifying theory. *J. Fluid Mech.* **407**, 27–56.

Hartmann, D.L., Moy, L.A. & Fu, Q. 2001 Tropical convection and the energy balance at the top of the atmosphere. *J. Clim.* **14** (24), 4495–4511.

Hinton, G. & Salakhutdinov, R. 2006 Reducing the dimensionality of data with neural networks. *Science* **313**(5786), 504–507,

Hopf, E. 1948 A mathematical example displaying features of turbulence. *Commun. Pure Appl. Maths* **1** (4), 303–322.

Hyman, J.M. & Nicolaenko, B. 1986 The Kuramoto–Sivashinsky equation: a bridge between PDE'S and dynamical systems. *Physica D: Nonlinear Phenom.* **18** (1), 113–126.

Jing, L., Zbontar, J. & Lecun, Y. 2020 Implicit rank-minimizing autoencoder. In *Advances in Neural Information Processing Systems* (ed. H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan & H. Lin), vol. 33, pp. 14736–14746. Curran Associates.

Kooloth, P., Sondak, D. & Smith, L.M. 2021 Coherent solutions and transition to turbulence in two-dimensional Rayleigh–Bénard convection. *Phys. Rev. Fluids* **6** (1), 013501.

Linot, A.J. & Graham, M.D. 2020 Deep learning to discover and predict dynamics on an inertial manifold. *Phys. Rev. E* **101** (6), 062209.

Linot, A.J. & Graham, M.D. 2022 Data-driven reduced-order modeling of spatiotemporal chaos with neural ordinary differential equations. *Chaos: Interdisciplinary J. Nonlinear Sci.* **32** (7), 073110.

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press

LINOT, A.J. & GRAHAM, M.D. 2023 Dynamics of a data-driven low-dimensional model of turbulent minimal Couette flow. *J. Fluid Mech.* **973**, A42.

PAGE, J., BRENNER, M.P. & KERSWELL, R.R. 2021 Revealing the state space of turbulence using machine learning. *Phys. Rev. Fluids* **6** (3), 034402.

PAGE, J., HOLEY, J., BRENNER, M.P. & KERSWELL, R.R. 2023 Exact coherent structures in two-dimensional turbulence identified with convolutional autoencoders. arXiv:2309.12754.

PANDEY, S., TEUTSCH, P., MÄDER, P. & SCHUMACHER, J. 2022 Direct data-driven forecast of local turbulent heat flux in Rayleigh–Bénard convection. *Phys. Fluids* **34** (4), 045106.

PEDREGOSA, F., et al. 2011 Scikit-learn: machine learning in python. *J. Machine Learning Res.* **12**, 2825–2830.

SCHEEL, J.D., EMRAN, M.S. & SCHUMACHER, J. 2013 Resolving the fine-scale structure in turbulent Rayleigh–Bénard convection. *New J. Phys.* **15** (11), 113063.

SIGGIA, E. 1994 High Rayleigh number convection. *Annu. Rev. Fluid Mech.* **26** (1), 137–168.

SONDAK, D. & PROTOPAPAS, P. 2021 Learning a reduced basis of dynamical systems using an autoencoder. *Phys. Rev. E* **104** (3), 034202.

SUGIYAMA, K., NI, R., STEVENS, R.J., CHAN, T. S., ZHOU, S.-Q., XI, H.-D., SUN, C., GROSSMANN, S., XIA, K.-Q. & LOHSE, D. 2010 Flow reversals in thermally driven turbulence. *Phys. Rev. Lett.* **105** (3), 034503.

TEMAM, R. 1990 Inertial manifolds. *Math. Intell.* **12** (4), 68–74.

VÅGE, K., PAPRITZ, L., HÅVIK, L., SPALL, M.A. & MOORE, G. 2018 Ocean convection linked to the recent ice edge retreat along east Greenland. *Nat. Commun.* **9** (1), 1287.

VAN DER, P., ERWIN, P., STEVENS, R.J., SUGIYAMA, K. & LOHSE, D. 2012 Flow states in two-dimensional Rayleigh–Bénard convection as a function of aspect-ratio and Rayleigh number. *Phys. Fluids* **24** (8), 085104.

WILLIS, A.P., SHORT, K.Y. & CVITANOVIĆ, P. 2016 Symmetry reduction in high dimensions, illustrated in a turbulent pipe. *Phys. Rev. E* **93** (2), 022204.

ZENG, K. & GRAHAM, M.D. 2023 Autoencoders for discovering manifold dimension and coordinates in data from complex dynamical systems. arXiv:2305.01090.

ZHU, X., MATHAI, V., STEVENS, R.J., VERZICCO, R. & LOHSE, D. 2018 Transition to the ultimate regime in two-dimensional Rayleigh–Bénard convection. *Phys. Rev. Lett.* **120** (14), 144502.

https://doi.org/10.1017/jfm.2025.21 Published online by Cambridge University Press