



PAPER

OPEN ACCESS

RECEIVED
13 October 2025

REVISED
4 February 2026

ACCEPTED FOR PUBLICATION
23 February 2026

PUBLISHED
17 April 2026

Original content from
this work may be used
under the terms of the
Creative Commons
Attribution 4.0 licence.

Any further distribution
of this work must
maintain attribution to
the author(s) and the title
of the work, journal
citation and DOI.



Application of deep neural networks for computing the renormalization group flow of the two-dimensional ϕ^4 field theory

Yueqi Zhao , Michael M Fogler and Yi-Zhuang You*

Department of Physics, University of California San Diego, La Jolla, CA 92093, United States of America

* Author to whom any correspondence should be addressed.

E-mail: yyou@ucsd.edu

Keywords: renormalization group, neural networks, ϕ^4 theory, critical exponents, normalizing flows

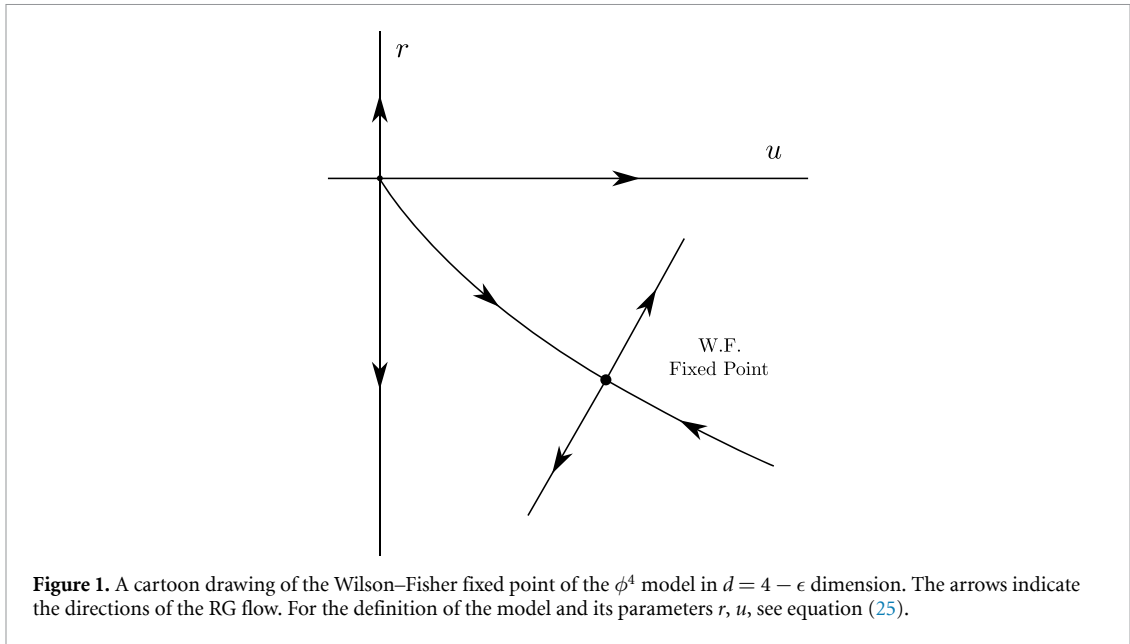
Abstract

We introduce RGFlow, a deep neural network–based real-space renormalization group (RG) framework tailored for continuum scalar field theories. Leveraging generative capabilities of flow-based neural networks, RGFlow autonomously learns real-space RG transformations from data without prior knowledge of the underlying model. In contrast to conventional approaches, RGFlow is bijective (information-preserving) and is optimized based on the principle of minimal mutual information. We demonstrate the method on two examples. The first one is a one-dimensional Gaussian model, where RGFlow is shown to learn the classical decimation rule. The second is the two-dimensional ϕ^4 theory, where the network successfully identifies a Wilson–Fisher-like critical point and provides an estimate of the correlation-length critical exponent.

1. Introduction

The renormalization group (RG) framework is one of the cornerstones of modern theoretical physics, a powerful tool for analyzing systems with interacting degrees of freedom (DOFs). It is typically formulated for effective actions of the form $\mathcal{S}(\mathbf{K})$, where $\mathbf{K}$ denotes a set of coupling constants and the theory includes an ultraviolet (UV) cutoff. In this work, we consider real-space RG transformations, in which the cutoff corresponds to the minimal spatial scale—such as the nearest-neighbor distance a on a lattice. The RG formalism describes how the parameters $\mathbf{K}$ evolve as this scale is varied. Over the decades, a range of techniques have been devised to implement such transformations. Among them, the block decimation method for spin systems is widely used: the system is divided into local clusters, and rules like majority voting or simple averaging are applied within each block to define new coarse-grained variables. This leads to an RG map $\mathbf{K}' = \mathcal{R}(\mathbf{K})$ and an effective rescaling of the lattice constant to $a' = ba$, with $b > 1$. Of particular interest are the fixed points of this map, which correspond to phases if these fixed points are stable or phase transitions if they are unstable. For example, the celebrated Wilson–Fisher fixed point (figure 1) describes the Ising-like ferromagnetic transition of the ϕ^4 field theory.

However, traditional RG schemes have some limitations. Their effectiveness often depends on the choice of block structure and averaging rules. While increasing cluster size can improve accuracy, it also makes the RG transformation more complex and analytically intractable. Moreover, designing an RG flow that captures the key physics requires significant human intuition—one must often guess the relevant order parameters in advance and construct rules tailored to them. For example, the naive block-spin majority-vote RG scheme does not directly apply to antiferromagnetic systems, as the coarse-graining rule must be adapted to account for the underlying antiferromagnetic order in order to preserve the relevant ordering at larger scales. Therefore, it is desirable to develop automated, data-driven RG approaches that do not rely on human intuition to construct coarse-graining rules, but can instead learn them directly from microscopic configurations sampled from models at different length scales.



Recently, deep neural networks (DNNs) have emerged as powerful tools for automating complex tasks across a wide array of scientific disciplines. Applications range from data interpretation and feature extraction in microscopy [3, 6, 19, 31, 54, 55] to the design and optimization of photonic or electronic devices [7, 16, 20, 22]. In statistical physics, DNNs have also been applied to design effective RG transformations at the configuration level. A central question in this line of work is to identify a guiding principle that defines the optimal RG rule. Koch-Janusz and Ringel [29] and Lenggenhager *et al* [33] proposed that the RG transformation should maximize the mutual information between the coarse-grained DOFs and their environment. Hu *et al* [24] put forward a complementary perspective, advocating that the RG transformation should instead minimize the mutual information among the irrelevant (decimated) DOFs. These principles have been implemented using a variety of architectures: flow-based models [2, 46], restricted Boltzmann machines [8], and related variational approaches [18, 40], as well as diffusion models [45]. Many of these works focus on finding the optimal RG transformation at the configuration level rather than on the flow of model parameters in coupling space. Recent studies by Di Sante *et al* [14], Ueda and Oshikawa [48], and Hou and You [23] have begun to explore RG flows in systems with discrete variables (spins). Here we extend this direction to continuum scalar field theories by combining the minimal mutual information principle with flow-based generative models to compute the RG flow of coupling constants directly.

In this work, we present RGFlow, a DNN-based real-space RG framework tailored to continuum scalar field theories. Our approach learns transformation rules from data and generates RG flow diagrams that encapsulate the model’s fixed points and critical exponents. The structure of this paper is as follows. Section 2 introduces the architecture of RGFlow and details the network design. In section 3, we consider the RGFlow optimization problem for a one-dimensional (1D) Gaussian model, which is analytically solvable. Section 4 demonstrates the application of RGFlow to the two-dimensional (2D) ϕ^4 theory, revealing a Wilson–Fisher-like critical point (figure 1) and estimating associated exponents. We end with concluding remarks in section 5.

2. RGFlow architecture

Consider a lattice field theory defined on a d -dimensional cubic lattice. Let $\phi(\mathbf{x}) \in \mathbb{R}$ denote a scalar field configuration sampled from the fine-grained action $\mathcal{S}_{UV}[\phi(\mathbf{x}); \mathbf{K}_{UV}]$, which is defined on a lattice with $(2N)^d$ sites (with $\mathbf{x} = (x_1, x_2, \dots, x_d)$ coordinating the lattice site). Likewise, let $\psi(\mathbf{x})$ denote a configuration sampled from the coarse-grained action $\mathcal{S}_{IR}[\psi(\mathbf{x}); \mathbf{K}_{IR}]$, defined on a coarser lattice with N^d sites. Here, $\mathbf{K}_{UV}$ and $\mathbf{K}_{IR}$ represent the sets of coupling constants at the UV and infrared (IR) scales, respectively.

The probability density functions associated with these field configurations are given by the Boltzmann distributions (assuming unit temperature):

$$\begin{aligned} P_{\text{UV}}[\phi] &= \frac{e^{-S_{\text{UV}}[\phi; \mathbf{K}_{\text{UV}}]}}{Z_{\text{UV}}}, \\ P_{\text{IR}}[\psi] &= \frac{e^{-S_{\text{IR}}[\psi; \mathbf{K}_{\text{IR}}]}}{Z_{\text{IR}}}, \end{aligned} \quad (1)$$

where $Z_{\text{UV/IR}} = \sum_{[\phi]/[\psi]} e^{-S_{\text{UV/IR}}}$ represents the partition function. From a probabilistic point of view, a general RG transformation can be described using the conditional probability $P_{\theta}[\psi|\phi]$ where θ denotes the set of parameters that parametrize the local transformation (coarse-graining) rule, such that the coarse-grained (IR) and fine-grained (UV) probability density functions are related by the integral

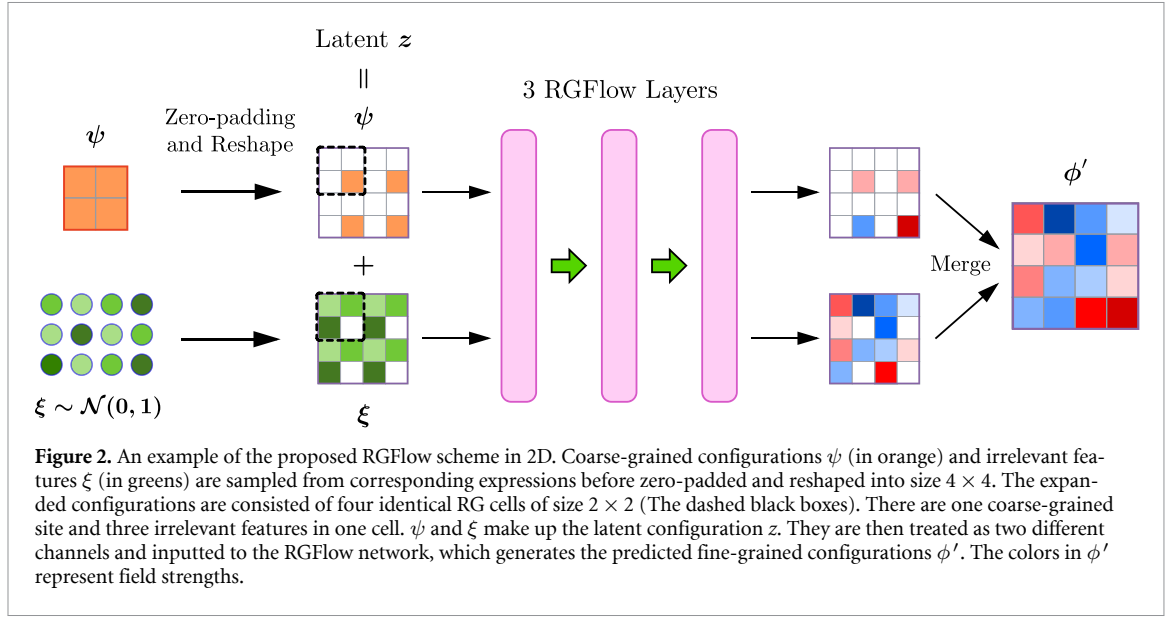
$$P_{\text{IR}}[\psi] = \int \mathcal{D}\phi \, P_{\theta}[\psi|\phi] P_{\text{UV}}[\phi]. \quad (2)$$

It is evident that there is substantial freedom in the choice of RG transformations as parameterized by some θ , and most choices do not yield physically meaningful predictions. Traditionally, RG transformations are optimized by imposing constraints on the physical properties of the transformed action. For instance, one class of methods computes magnetic scaling exponents independently from both the free energy and the correlation functions, and then enforces their consistency [26, 50]. Another approach involves optimizing the variational lower and upper bounds on the free energy prior to the transformation [25, 35, 36]. However, these methods typically involve a limited number of tunable parameters, due to the computational complexity and analytical effort required. More recently, optimization principles based on information theory have been introduced. Unlike the traditional approaches, these methods directly optimize probability densities, therefore substantially reducing computational overhead. This advancement enables the RG transformation to be represented by much more complex models, such as restricted Boltzmann machines [29, 33], or even neural networks [24].

In this study, we adopt the minimal bulk mutual information principle formulated previously in [24]. This principle posits that the RG transformation should eliminate irrelevant features—defined as those DOFs that exhibit minimal mutual information. In the ideal case, this minimum is achieved when the decimated DOFs are mutually independent and behave as decoupled random noise. The motivation for this principle comes from the well-known fact that mutual information provides an upper bound on statistical correlation functions (see appendix A). By minimizing the mutual information among the discarded DOFs, the optimized RG transformation effectively suppresses irrelevant correlations and compresses the relevant physical information into the coarse-grained variables. Furthermore, [24] establishes the equivalence between the minimal bulk mutual information principle and the more widely studied maximal real-space mutual information principle [29, 33], which aims to maximize the mutual information between the coarse-grained field and its surrounding fine-grained environment. In our formulation, we model the irrelevant features [denoted by $\xi(\mathbf{x})$] as independent Gaussian random variables with distribution $P[\xi] = \mathcal{N}(0, \mathbb{1})$, so that the minimal bulk mutual information criterion is satisfied by design.

Additionally, we have chosen our RG map to be bijective. Instead of discarding the irrelevant features, as in traditional RG, we keep them. The coarse-grained configuration and the generated random noises then jointly form a ‘latent configuration,’ which shares the same dimension as the fine-grained one. Such kind of information-preserving RG transformation has been successfully applied to a variety of field theory models [32, 52, 53]. Finally, we implement our bijective RGFlow network using so-called normalizing flow architecture [28, 38, 39], whose network parameters become the aforementioned set of parameters θ characterizing the RG transformation. The optimality is achieved through training the neural network.

The proposed RGFlow renormalization scheme utilizes a local transformation rule in the inverse renormalization direction and generates ϕ from ψ in the following three steps. (I) Sample a coarse-grained configuration ψ and a vector ξ of irrelevant features of size $(2N)^d - N^d$ from $P_{\text{IR}}[\psi]$ and $P[\xi]$ respectively. (II) ψ and ξ are expanded using zero-padding and reshaped into size $(2N)^d$. The resulting configuration consists of N^d RG cells of 2^d sites. Each RG cell contains one lattice site from ψ and $2^d - 1$ sites from ξ . (III) This configuration is given as an input to the network, which generates a predicted fine-grained configuration $\phi'(\mathbf{x})$. The rescaling factor for our RG scheme is $b = a'/a = 2$ since the configuration size has doubled following this procedure. The above RG steps are depicted in figure 2 for the case $d=2$. It is important to notice that our RGFlow scheme is similar to the NeuralRG [46]. Both architectures treat the irrelevant DOFs as mutually independent Gaussian noises and use a neural network to approximate the bijective RG transformation. Further, the RGFlow model could be viewed



as the first layer in the hierarchical NeuralRG structure. However, there still exists a major difference. In the NeuralRG model, the fine-grained configuration is mapped entirely to a random noise, whereas in the RGFlow scheme, we still keep track of the coarse-grained coupling constants $\mathbf{K}_{\text{IR}}$ and therefore their renormalization.

To further simplify the notation, we define $z = (\psi, \xi)$ to be our latent configuration and we denote the RG transformation by $z = T_\theta(\phi)$ where θ are the network parameters. We can express the model distribution $P'_{\text{UV}}[\phi]$ for the RGFlow predicted fine-grained configurations ϕ' using $P_{\text{IR}}[\psi]$ and $P[\xi]$ as

$$P'_{\text{UV}}[\phi] = P[z] / \det J_{T^{-1}}(z), \quad J_{T^{-1}}(z) = \frac{\delta T_\theta^{-1}(z)}{\delta z}. \quad (3)$$

Here $J_{T^{-1}}(z)$ is the Jacobian matrix of the transformation and $P[z]$ is the probability distribution function of z :

$$P[z] = P_{\text{IR}}[\psi] P[\xi]. \quad (4)$$

Using equations (3) and (4) we can explicitly express the conditional probability in equation (2) as

$$P_\theta[\psi|\phi] = \frac{\det J_{T^{-1}}(z)}{P[\xi]} \delta(T_\theta^{-1}(z) - \phi). \quad (5)$$

Our goal is to construct T_θ such that the generated fine-grained configurations $\phi'(\mathbf{x})$ have the same probability distribution as true ϕ . This requires maximizing the similarity between $P'_{\text{UV}}[\phi]$ and the ideal distribution $P_{\text{UV}}[\phi]$ from equation (1). A popular measure of the similarity between two probability distributions is the Kullback–Leibler (KL) divergence. However, since the partition function Z_{UV} and Z_{IR} are intractable for most lattice field theories, direct evaluation of the KL divergence is impossible. Therefore, we instead choose the Fisher divergence, which takes the following form:

$$\begin{aligned} D_F(P'_{\text{UV}} || P_{\text{UV}}) &= \mathbb{E}_{\phi \sim P'_{\text{UV}}[\phi]} \left\| \frac{\delta \log P'_{\text{UV}}[\phi]}{\delta \phi} - \frac{\delta \log P_{\text{UV}}[\phi]}{\delta \phi} \right\|_2^2 \\ &= \mathbb{E}_{\phi \sim P'_{\text{UV}}[\phi]} \left\| \frac{\delta \mathcal{S}'_{\text{UV}}[\phi; \mathbf{K}_{\text{IR}}]}{\delta \phi} - \frac{\delta \mathcal{S}_{\text{UV}}[\phi; \mathbf{K}_{\text{UV}}]}{\delta \phi} \right\|_2^2. \end{aligned} \quad (6)$$

Equation (6) can be easily evaluated since we have from equations (3) and (4)

$$\mathcal{S}'_{\text{UV}}[\phi; \mathbf{K}_{\text{IR}}] = \mathcal{S}_z[z; \mathbf{K}_{\text{IR}}] + \log \det J_{T^{-1}}(z), \quad (7)$$

$$\mathcal{S}_z[z; \mathbf{K}_{\text{IR}}] = \mathcal{S}_{\text{IR}}[\psi; \mathbf{K}_{\text{IR}}] + \frac{1}{2} \xi^T \xi, \quad (8)$$

where $\mathcal{S}_z[z; \mathbf{K}_{\text{IR}}]$ is the latent action. The Jacobian matrix of the RG transformation is $J_T(\phi) = \frac{\delta z}{\delta \phi} = \frac{\delta T_\theta(\phi)}{\delta \phi}$. Applying the chain rule for derivatives, we transform the Fisher divergence to

$$D_F(P'_{\text{UV}} || P_{\text{UV}}) = \mathbb{E}_{z \sim P_z[z]} \left\| \left[\frac{\delta(\mathcal{S}_z[z; \mathbf{K}_{\text{IR}}] + \log \det J_{T^{-1}}(z))}{\delta z} - \frac{\delta \mathcal{S}_{\text{UV}}[T_\theta^{-1}(z); \mathbf{K}_{\text{UV}}]}{\delta z} \right] J_T(\phi) \right\|_2^2. \quad (9)$$

Since for a meaningful RG map $J_T(\phi)$ should not vanish, we drop this factor and define our training loss function to be

$$L(\mathbf{K}_{\text{IR}}, \theta) = \mathbb{E}_{z \sim P_z[z]} \left\| \left[\frac{\delta(\mathcal{S}_z[z; \mathbf{K}_{\text{IR}}] + \log \det J_{T^{-1}}(z))}{\delta z} - \frac{\delta \mathcal{S}_{\text{UV}}[T_\theta^{-1}(z); \mathbf{K}_{\text{UV}}]}{\delta z} \right] \right\|_2^2. \quad (10)$$

This loss function has the same global minimum (zero) as D_F . Evaluating $L(\mathbf{K}_{\text{IR}}, \theta)$ requires computing the Jacobian $\det J_{T^{-1}}(z)$. To make it computationally efficient, we implement the mapping T_θ using several RealNVP modules [15]. The corresponding Jacobian matrix is triangular, and so its determinant is equal to the product of the diagonal elements of $J_{T^{-1}}(z)$. To utilize the RealNVP module, we treat the zero-padded ψ and ξ as two input channels. After passing each layer, the order of the two channels switch. Therefore, two consecutive RealNVP modules make up one RGFlow layer, which functions as a basic building block of the network. As demonstrated in the 2D example in figure 1, the upper channel contains only the coarse-grained fields ψ while the lower channel holds the remaining irrelevant features. After being transformed by the neural network, the two channels are merged to reconstruct the predicted ϕ' .

During training, a batch of configurations (ψ, ξ) are sampled from $P_{\text{IR}}[\psi]$ and $P[\xi]$ to calculate $L(\mathbf{K}_{\text{IR}}, \theta)$ at each step. A stochastic optimizer is then applied to compute the gradients and optimize T_θ^{-1} and $\mathbf{K}_{\text{IR}}$ simultaneously. After reaching a local optimum, a unique set of predicted coarse-grained coupling $\mathbf{K}_{\text{IR}}$ will be associated with the fined-grained couplings $\mathbf{K}_{\text{UV}}$, which are kept fixed during training. By separately probing different sets of $\mathbf{K}_{\text{UV}}$, we can study the RG flow at any location in the parameter space. In other words, we do not impose any limitations on the coupling constants or space dimension (unlike, e.g. the ϵ -expansion [17, 51]). Moreover, since a unique neural network is trained for every pair of $\mathbf{K}_{\text{UV}}$ and $\mathbf{K}_{\text{IR}}$, the learned RG map is automatically adjusted to extract the most relevant feature. Therefore, we expect that the proposed method may be suitable for studying the RG flows of systems with multiple phases.

3. A solvable 1D model

The described RGFlow scheme can be solved exactly for the case of a 1D Gaussian model with the action

$$\mathcal{S}_{\text{UV}}[\phi; r_{\text{UV}}] = \frac{1}{2} \sum_{j=0}^{2N-1} \left[(\phi_{j+1} - \phi_j)^2 + r_{\text{UV}} \phi_j^2 \right], \quad (11)$$

where ϕ_j is a scalar field on j th site of an $2N$ -site lattice with the periodic boundary conditions and a unit lattice spacing constant. It is convenient to double the unit cell to match the latent model, see below, and introduce the two-component fields $\Phi_k = (\phi_{2k}, \phi_{2k+1})^T$, where $k = 0, 1, \dots, N-1$. We define the Fourier transform $\tilde{\Phi}(q) = N^{-1/2} \sum_{k=0}^{N-1} e^{-iqk} \Phi_k$ of these fields and rewrite the model action as follows:

$$\mathcal{S}_{\text{UV}} = \frac{1}{2} \sum_{n=0}^{N-1} \tilde{\Phi}(q_n)^\dagger \Omega_{\text{UV}}(q_n) \tilde{\Phi}(q_n), \quad \Omega_{\text{UV}}(q_n) = \begin{pmatrix} r_{\text{UV}} + 2 & -(1 + e^{-iq_n}) \\ -(1 + e^{iq_n}) & r_{\text{UV}} + 2 \end{pmatrix}, \quad q_n = \frac{2\pi n}{N}. \quad (12)$$

Note that the eigenvalues $\omega_\pm(q) = r_{\text{UV}} + 2 \pm 2 \cos q$ of the matrix $\Omega_{\text{UV}}(q)$ represent the band dispersion in the problem.

In turn, the action of the latent model is

$$\mathcal{S}_Z = \frac{1}{2} \sum_{j=0}^{N-1} \left\{ \left[(\psi_{j+1} - \psi_j)^2 + r_{\text{IR}} \psi_j^2 \right] + \xi_j^2 \right\}. \quad (13)$$

We again introduce two-components fields $Z_j = (\psi_j, \xi_j)^T$ and the corresponding Fourier transforms $\tilde{Z}(q_n)$, in terms of which the latent action becomes

$$\mathcal{S}_Z = \frac{1}{2} \sum_{n=0}^{N-1} \tilde{Z}(q_n)^\dagger \Omega_Z(q_n) \tilde{Z}(q_n), \quad \Omega_Z(q_n) = \begin{pmatrix} r_{\text{IR}} + 2 - 2\cos q_n & 0 \\ 0 & 1 \end{pmatrix}. \quad (14)$$

Since both $\mathcal{S}_{\text{UV}}$ and $\mathcal{S}_Z$ are quadratic, the exact global minimum $L=0$ of the loss function L from equation (10) can be achieved by a linear RG transformation T_θ of a convolution type such that

$$\tilde{Z}(q) = \tilde{T}_\theta(q) \tilde{\Phi}(q), \quad (\tilde{T}_\theta)^\dagger \Omega_Z \tilde{T}_\theta = \Omega_{\text{UV}}. \quad (15)$$

Here, as above, the tilde denotes the Fourier transform, which is defined as follows. It suffices to consider T_θ with the following general size-3 convolution kernels for ψ 's and ξ 's:

$$\mathcal{C}_\psi = \{a, c, d\}, \quad \mathcal{C}_\xi = \{f, g, h\} \quad (16)$$

with real a, c, d, f, g and h . Matrix $\tilde{T}_\theta$ is then given by

$$\tilde{T}_\theta(q_n) = \begin{pmatrix} a + e^{iq_n} d & c \\ e^{iq_n} g & f + e^{iq_n} h \end{pmatrix}. \quad (17)$$

Equation (15) has the exact solution $a = d = 0$, $c = f = h = -1/g$, and $g = \pm\sqrt{r_{\text{UV}} + 2}$. The corresponding renormalized coupling is

$$r_{\text{IR}} = 4r_{\text{UV}} + r_{\text{UV}}^2. \quad (18)$$

The first term on the right-hand side of this equation agrees with the scaling law $r_{\text{IR}} = b^2 r_{\text{UV}}$ known to apply in the continuum limit. The second term can be considered a lattice correction. Importantly, this lossless RG scheme preserves the behavior of the correlation functions, such as the exponential decay of the two-point function

$$\langle \phi_m \phi_n \rangle = \int_{-\pi}^{\pi} \frac{dq}{2\pi} \frac{e^{i(m-n)q}}{r_{\text{UV}} + 2 - 2\cos q} = \frac{1}{2 \sinh \varkappa_{\text{UV}}} e^{-\varkappa_{\text{UV}} |m-n|}. \quad (19)$$

(This formula applies for the infinite lattice, $N \rightarrow \infty$.) Parameter

$$\varkappa_{\text{UV}} = \text{arccosh} \left(1 + \frac{1}{2} r_{\text{UV}} \right) \quad (20)$$

is the inverse correlation length. Equation (18) entails $\varkappa_{\text{IR}} = 2\varkappa_{\text{UV}}$, so that the two-point functions before and after the RG transformation are related by

$$\langle \psi_0 \psi_i \rangle = Z \langle \phi_0 \phi_{2i} \rangle, \quad (21)$$

as required for the case $b=2$ we are dealing with. This relation can be also seen directly from the form of $\mathcal{C}_\psi$ kernel; in particular, the coefficient of proportionality in equation (21), i.e. the field renormalization factor is simply $Z = c^2 = (r_{\text{UV}} + 2)^{-1}$.

The above results are also in agreement with those derived from the standard real-space RG procedure employing decimation. In such an approach, the Gaussian chain is partitioned into unit cells each containing one even-indexed site and one odd-indexed site, $\Phi_j = (\phi_{e,j}, \phi_{o,j})$, where $j = 0, 1, \dots, N-1$. Under this representation, the Gaussian action (11) can be expressed as follows:

$$\mathcal{S}_{\text{UV}}[\phi; r_{\text{UV}}] = \sum_{j=0}^{N-1} \mathcal{S}_j = \sum_{j=0}^{N-1} \frac{1}{2} (\phi_{o,j} - \phi_{e,j})^2 + \frac{1}{2} (\phi_{e,j+1} - \phi_{o,j})^2 + \frac{r_{\text{UV}}}{2} \left(\frac{1}{2} \phi_{e,j}^2 + \frac{1}{2} \phi_{e,j+1}^2 + \phi_{o,j}^2 \right). \quad (22)$$

After completing the square with respect to $\phi_{o,j}$, we can separate $\mathcal{S}_j$ into two parts

$$\begin{aligned} \mathcal{S}_j &= \mathcal{S}_{\text{N},j}[\xi] + \mathcal{S}_{\text{IR},j}[\psi; r_{\text{IR}}]. \\ &= \frac{1}{2} \xi_j^2 + \frac{1}{2} \left[(\psi_{j+1} - \psi_j)^2 + \frac{1}{2} r_{\text{IR}} (\psi_j^2 + \psi_{j+1}^2) \right], \end{aligned} \quad (23)$$

with the following definition

$$\begin{aligned}\xi_j &= -\frac{1}{\sqrt{2+r_{UV}}}\phi_{e,j} + \sqrt{2+r_{UV}}\phi_{o,j} - \frac{1}{\sqrt{2+r_{UV}}}\phi_{e,j+1}, \\ \psi_j &= \frac{1}{\sqrt{2+r_{UV}}}\phi_{e,j}, \quad \psi_{j+1} = \frac{1}{\sqrt{2+r_{UV}}}\phi_{e,j+1}, \\ r_{IR} &= 4r_{UV} + r_{UV}^2.\end{aligned}\tag{24}$$

It is evident that $\mathcal{S}_{N,j}[\xi]$ corresponds to unit Gaussian noise and $\mathcal{S}_{IR,j}[\psi; r_{IR}]$ is consistent with the Gaussian action (11) under periodic boundary conditions. In addition, the definitions provided in (24) reproduce both the kernel parameters and the scaling relations obtained by minimizing the loss function (15). This correspondence allows for a natural interpretation of the kernel $\mathcal{C}_\psi$ as performing a decimation operation, effectively selecting every second site in the fine-grained configuration. Simultaneously, the kernel $\mathcal{C}_\xi$ captures the discarded DOFs and encodes them as Gaussian noise. In conventional real-space RG, the transformation is completed by integrating out the Gaussian noise $\mathcal{S}_{N,j}[\xi]$, thereby making the procedure non-invertible. In contrast, the RGFlow framework retains this information, thus preserving bijectivity and enabling a fully reversible transformation.

4. Numerical results

4.1. Model

We now focus on the 2D ϕ^4 theory on a square lattice with the periodic boundary conditions and action

$$\mathcal{S}[\phi; r, u] = \frac{1}{2} \sum_{\langle ij \rangle} (\phi_i - \phi_j)^2 + \sum_i \frac{r}{2} \phi_i^2 + \frac{u}{4} \phi_i^4,\tag{25}$$

where ϕ_i is a real valued scalar field defined on lattice site i and $\langle ij \rangle$ is the bond between nearest-neighbor lattice sites i, j . The coupling constants $\mathbf{K} = (u, r)$ are dimensionless. Constant u must be non-negative for stability.

At large u , the lattice ϕ^4 theory can be mapped onto the Ising model $S = -J \sum_{\langle ij \rangle} \sigma_i \sigma_j$ with the dimensionless exchange constant $J = r/u$ and discrete site variables $\sigma_i = \pm 1$. From Onsager's exact solution of the 2D Ising model we know that the system undergoes the ferromagnetic transition at

$$r \simeq -c_1 u, \quad c_1 = \frac{1}{2} \ln(1 + \sqrt{2}), \quad u \gg 1.\tag{26}$$

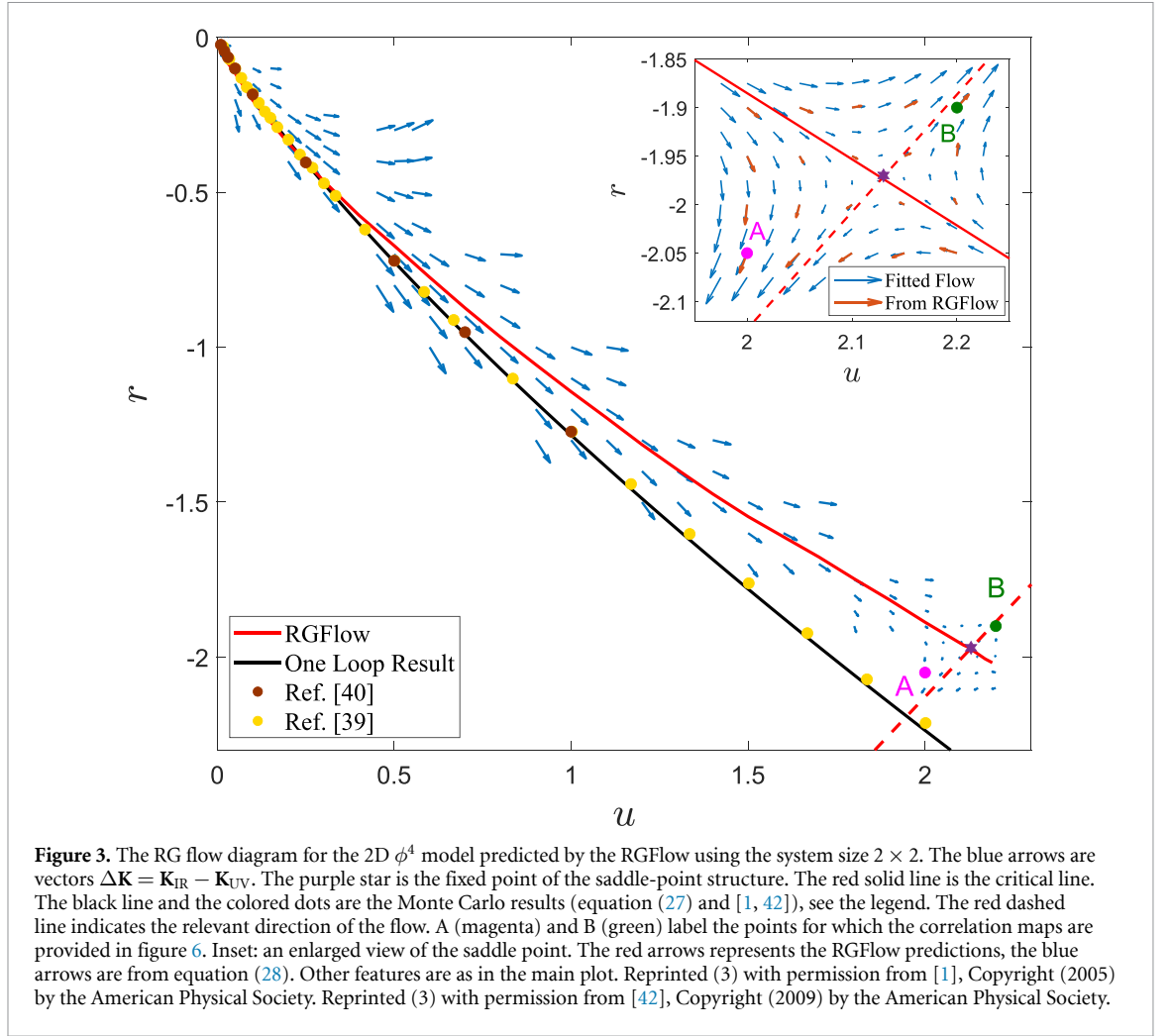
It is believed that at smaller u the transition remains within the Ising universality class. The corresponding critical line in the (u, r) parameter space has been studied by Monte-Carlo simulations [1, 34, 37, 42, 47] and other methods ([13, 41, 43], etc). For small u , these calculations predict the critical r to be

$$r \simeq \frac{u}{c_0} + \frac{3}{4\pi} u \ln\left(\frac{u}{32c_0}\right), \quad 0 < u \ll 1.\tag{27}$$

The second term in this equation comes from the one-loop renormalization of the mass term r on the square lattice. At $u \rightarrow 0$, most recent calculations predict $c_0 \approx 11$, see table 1 in [13]. However, using $c_0 = 9.9$ gives an excellent overall fit to the Monte-Carlo data [1, 34] within the range $0 < u < 2$, see figure 3.

It is easy to see that the phase-transition line approximated by equation (27) must be a streamline of the RG flow, in the following sense. According to the modern Wilsonian formulation of the RG, the exact RG flow involves infinite number of coupling constants besides r and u , e.g. those for ϕ^6, ϕ^8 , etc. terms in the action. On the (u, r) -plane all those additional coupling constants are equal to zero. Importantly, any streamline of the exact RG flow that starts from a point on the phase transition line in the (u, r) -plane eventually arrives at the same fixed point in the higher-dimensional space. The union of these streamlines defines the RG critical surface [44]. The projection of the flow vectors of this critical surface onto the (u, r) plane must be tangent to the phase-transition line. This reasoning holds as long as the coupling constants evolve continuously, which happens when the rescaling factor b approaches unity. In this work, we adopt $b = 2$, a common choice in real-space RG. Therefore, the RGFlow-computed critical line may somewhat deviate from the exact phase-transition line.

We expect that the critical line contains two fixed points, the Gaussian fixed point $u = r = 0$ and a nonperturbative (Wilson-Fisher-like) fixed point at $u = u^* > 0, r = r^* < 0$. The latter is the approximation of the projection of the true multi-dimensional fixed point onto the (u, r) plane. The RG flow



near (u^*, r^*) can be used to estimate the correlation length critical exponent ν [44]. Our goal is to compute the critical line by the RGFlow scheme to verify these expectations and to investigate any additional structures of the flow in a larger parameter space around the critical line.

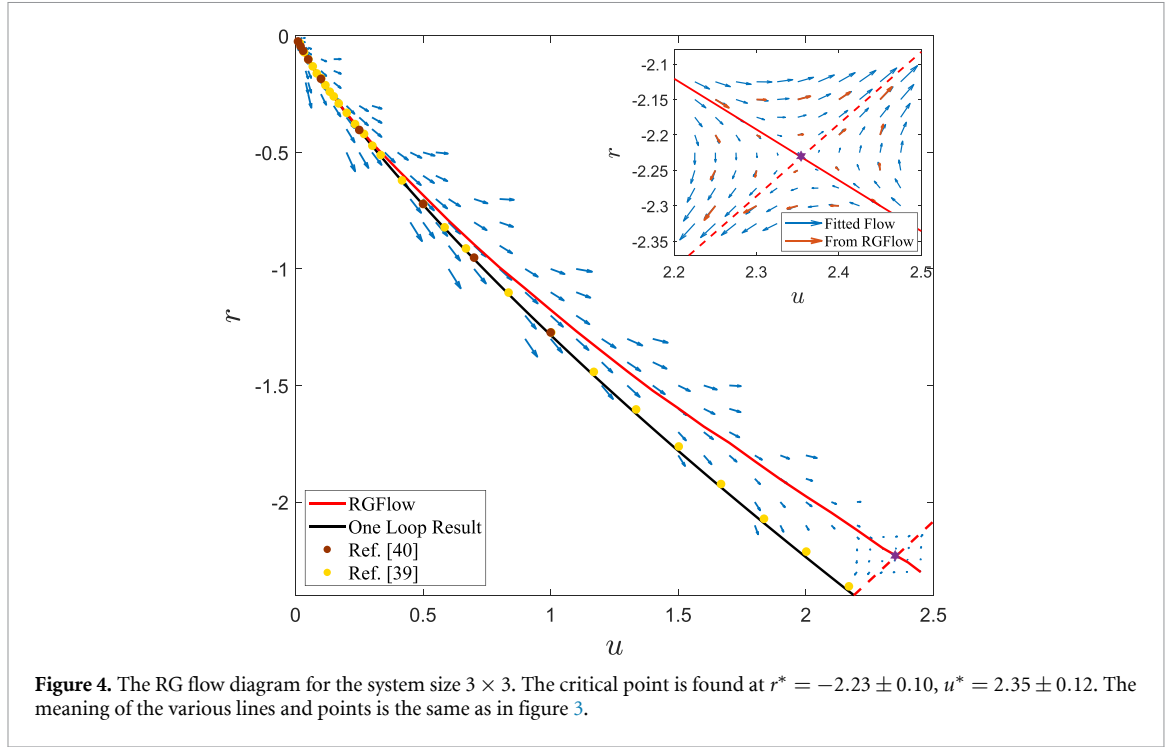
4.2. Critical line and critical exponents

To construct the RG map, we build a RealNVP module using 2D convolution layers with the kernel size of 3×3 . In total, three RGFlow layers are included in the neural network. The network structure is shown in figure 2. For a proof-of-principle demonstration of the proposed scheme, we chose a minimal system size of 2×2 for ψ . During training, a batch of 100 ψ and ξ configurations were sampled at each step. The Adam optimizer [27] with an initial step size of $h = 10^{-3}$ was applied to minimize the loss function (10). After every 5000 iterations, h was reduced by 85%. A total of 40000 training iterations were used for each session.

We applied the RGFlow algorithm to a set of points in the (u, r) plane near the expected position of the critical line. The results are displayed as a vector field $\Delta \mathbf{K} = \mathbf{K}_{\text{IR}} - \mathbf{K}_{\text{UV}}$ in figure 3. We observe three types of parameter flows: the first one goes to $r = \infty$ (the symmetric phase), the second one goes to $r = -\infty$ (the broken symmetry phase), the remaining third one flows into the saddle point (the transition). We find the location of this fixed point $\mathbf{K}^* = (u^*, r^*)$ as follows. We first apply the least squares method [11, 12] to fit the saddle point structure to a linear vector field

$$\Delta \mathbf{K}(\mathbf{K}_{\text{UV}}) = \mathbf{M} \mathbf{K}_{\text{UV}} + \mathbf{d}, \quad (28)$$

where $\mathbf{M}$ and $\mathbf{d}$ represent a 2×2 matrix and a 2×1 vector, respectively. Next, we compute $\mathbf{K}^* = -\mathbf{M}^{-1} \mathbf{d}$ such that $\Delta \mathbf{K}(\mathbf{K}^*) = 0$. In this numerical study, we find that the fixed point is located at $r^* = -1.97 \pm 0.12$ and $u^* = 2.13 \pm 0.13$ (the purple star in figure 3).



The critical line is defined to be the streamline of the vector field $\Delta \mathbf{K}$ that connects the Gaussian fixed point at the origin and the Wilson–Fisher-like fixed point $\mathbf{K}^*$. The result is displayed as the solid red curve in figure 3, which was computed using the `streamline` function of MATLAB. Fitting the RGFlow critical line to equation (27) at $0 < u < 2$, we obtain $c_0 = 6.44 \pm 0.06$, which is somewhat smaller than $c_0 = 9.9$ from the Monte–Carlo data (the black curve). However, this result was obtained from a tiny system size of 2×2 . It becomes more accurate once a larger system size is used, see below.

The critical exponents are estimated by linearizing the RG equation near the fixed point:

$$\begin{aligned} \mathbf{K}_{\text{IR}} - \mathbf{K}^* &= \left. \frac{\partial \mathbf{K}_{\text{IR}}}{\partial \mathbf{K}_{\text{UV}}} \right|_{\mathbf{K}^*} (\mathbf{K}_{\text{UV}} - \mathbf{K}^*) \\ &= (\mathbf{M} + \mathbb{1}) (\mathbf{K}_{\text{UV}} - \mathbf{K}^*). \end{aligned} \quad (29)$$

The Jacobian matrix $\mathbf{M} + \mathbb{1}$ of this linearized flow has two eigenvalues, λ_t and λ_h . Following the convention [4], we define $|\lambda_t| > 1$ as the relevant eigenvalue and its corresponding eigenvector to be the relevant direction. In turn, $|\lambda_h| < 1$ is the irrelevant eigenvalue and its corresponding eigenvector is the irrelevant direction. In figure 3, the relevant direction is along with the critical line while the irrelevant direction is along the dashed red line.

The critical exponent ν is calculated using

$$\nu = \frac{\log b}{\log |\lambda_t|}, \quad (30)$$

where $b = 2$. We obtain $\nu = 0.885 \pm 0.015$, which differs from the exact value of $\nu = 1$ by about 10%.

To test if the discrepancy in ν was due to the small system size, we did additional calculations for 3×3 systems. To achieve the training stability around the saddle point, we decreased the initial step size to $h = 10^{-4}$ and its reduction rate to 85% per 10^4 iterations. We also had to double the number of training iterations. The resulting flow diagram is presented in figure 4. As one can see from that figure and also from figure 5, the critical line indeed becomes closer to the Monte–Carlo results for this system size. The corresponding c_0 is given in table 1. The critical exponent ν also becomes closer to the exact result. Since the computing time has increased substantially by going from size 2×2 to 3×3 , treating even larger systems would probably require further optimization of the network design.

The remaining inaccuracy of ν , which is 4% for 3×3 , is presumably due to neglecting higher-order interactions ϕ^6 , etc. A similar situation has been reported in a recent study of a ML-based RG for the 2D Ising model co-authored by one of us [23]. By allowing for additional coupling terms in the UV

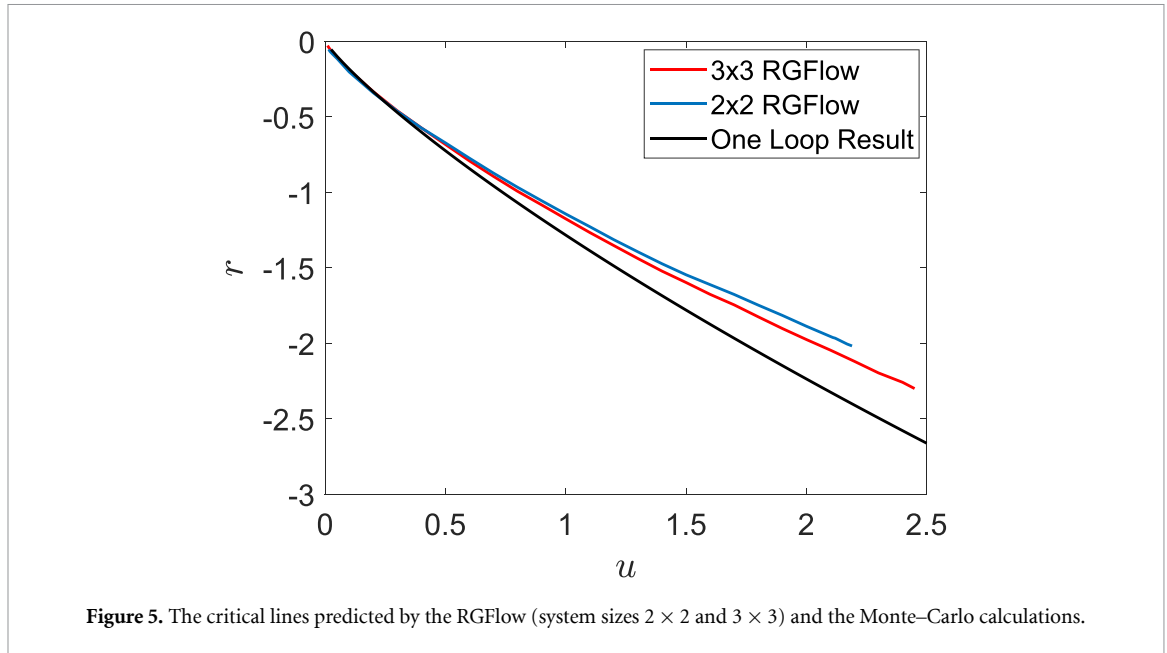


Figure 5. The critical lines predicted by the RGFlow (system sizes 2×2 and 3×3) and the Monte–Carlo calculations.

Table 1. Parameters predicted by the RGFlow vs. the exact results.

System size	2×2 .	3×3	Exact
c_0	6.44 ± 0.06	7.27 ± 0.06	9.9
ν	0.855 ± 0.015	0.958 ± 0.015	1

Hamiltonian, an improved prediction of the critical coupling constant was observed. We envision that a better estimation of ν can be achieved by similarly extending our UV action (25).

To reveal the RG rules learned by the neural network we calculate the normalized correlation functions

$$\begin{aligned} \langle \xi_i \phi_j \rangle &= \mathbb{E}_{\phi \sim P_{UV}[\phi]} \frac{\xi_i \phi_j}{\sigma(\xi_i) \sigma(\phi_j)}, \\ \langle \psi_i \phi_j \rangle &= \mathbb{E}_{\phi \sim P_{UV}[\phi]} \frac{\psi_i \phi_j}{\sigma(\psi_i) \sigma(\phi_j)} \end{aligned} \quad (31)$$

between a site i in the RG transformed configuration and all sites j in the fine-grained configuration ϕ (see figure 6(a)). Here $\sigma(x)$ stands for the standard deviation of x . For each variable ξ_i or ψ_i , the resulting correlation map is of size 4×4 and can be interpreted as a generalized convolutional kernel and, equivalently, a RG kernel. Since the transformed configurations are composed of repeated 2×2 RG cells (see figure 2), it is sufficient to evaluate the correlation maps for the four constituent sites within the top-left unit cell. To capture the phase dependence of the learned RG rules, we calculate the correlation maps at two representative points, labeled A and B, corresponding to the symmetric and symmetry-broken phases, respectively (see figure 3). The results are shown in figures 6(b) and (c).

In both phases, the correlation strengths between the coarse-grained site ψ_i and neighboring fine-grained fields ϕ_j remain roughly the same. This indicates that the RGFlow network has successfully identified a local averaging rule consistent with conventional real-space RG, which extracts long-range information. Notably, the strength of these correlations remains invariant across the phase transition, as expected for the ϕ^4 theory. In contrast, the correlation maps between the irrelevant features ξ and the surrounding ϕ_j exhibit patterns with opposite signs. These results suggest that the network has learned effective finite-difference-like kernels that extract short-range interactions. Importantly, the noise correlation maps associated with ξ remain phase independent and are consistent with the assumption that these irrelevant features follow a Gaussian distribution and are uncorrelated with the macroscopic phase behavior.

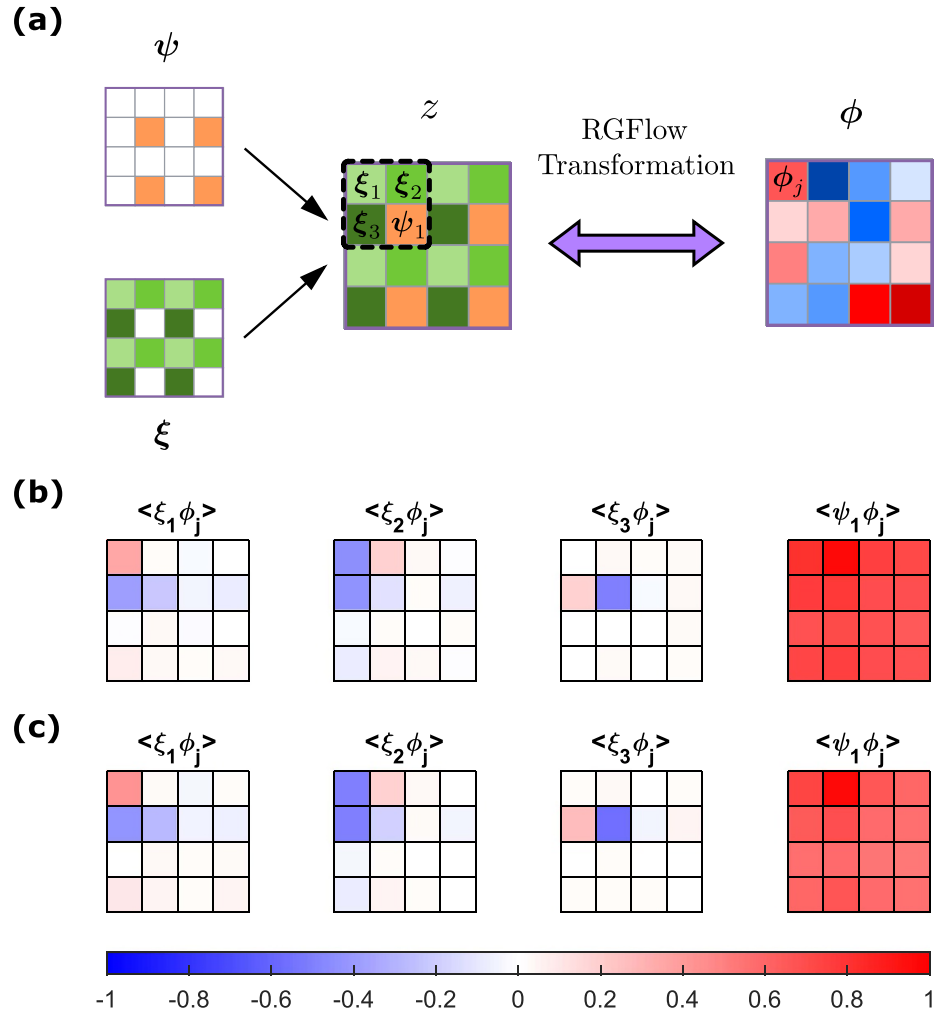
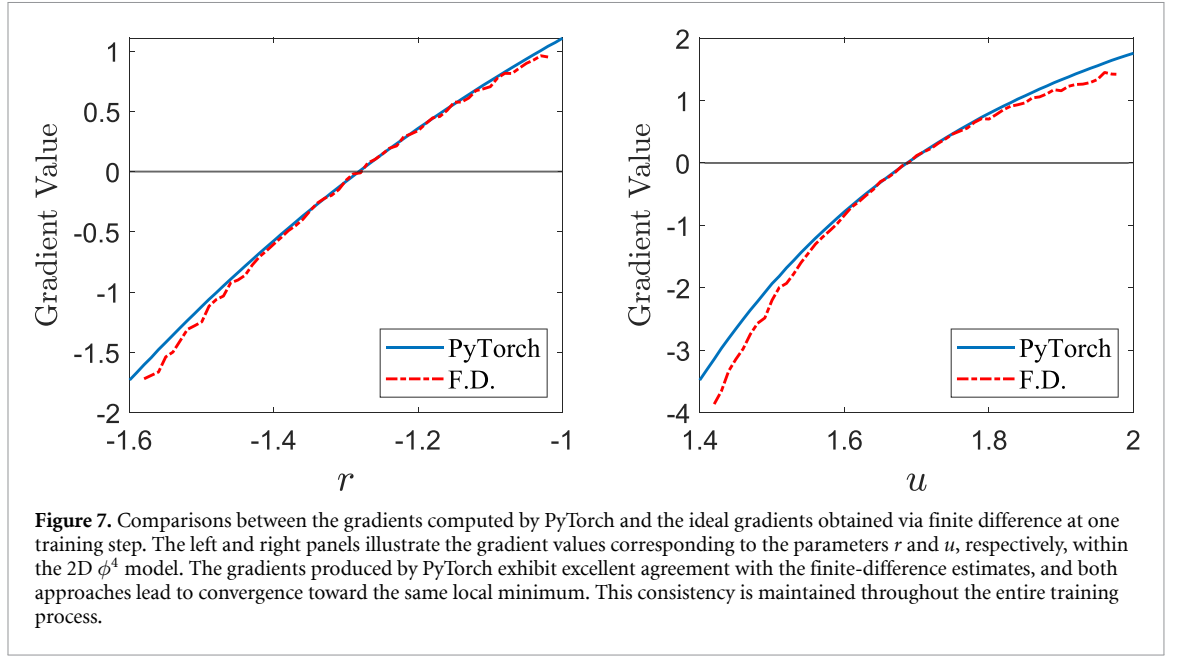


Figure 6. Correlation maps between sites in the RG unit cell and those in the fine-grained configuration. (a) The unit cell of interest (the dashed box) containing variables $\xi_1, \xi_2, \xi_3, \psi_1$ and the fine-grained unit cell containing variable ϕ_j . (b) The correlation maps (31) computed at the point A in figure 3. (c) Same as (b) for point B in figure 3.

5. Conclusions

In this study, we introduced RGFlow, a neural network-based framework for learning real-space RG transformations and parameter flows for lattice field theories. Unlike traditional RG approaches, RGFlow preserves information by employing a bijective transformation that maps fine-grained configurations to coarse-grained fields and irrelevant features. During training, we utilize the Fisher divergence as a loss function to circumvent the intractability of partition functions. We analytically validated the RGFlow algorithm using the 1D Gaussian model, demonstrating that it reproduces the correct parameter flow. Numerically, we applied RGFlow to the 2D ϕ^4 model. Remarkably, even when trained on the smallest possible 2×2 lattice configurations, the method still identified a saddle point structure in the parameter flow diagram and predicted the critical line and the associated critical exponent with only a $\sim 10\%$ error.

The RGFlow method demonstrates several advantages. First, it automates the design and optimization of RG transformations across a wide parameter space. Second, unlike existing neural-network-based RG methods for lattice field theories [2, 8, 18, 24, 29, 33, 40, 45, 46], RGFlow implementation does not require explicit knowledge of the physics of the model. This feature broadens its applicability to more complex field theories. Third, since the training criterion of RGFlow relies only on evaluating the actions, it is suitable for strongly coupled regimes, where traditional methods may struggle.



We can think of several directions for improvement and future investigations. Currently, RGFlow generates RG flows in a point-wise manner, which can be computationally intensive for complex theories. Inspired by the RG Monotone method [23], a promising enhancement would be to introduce a second neural network to extrapolate the parameter flow for a wide range of parameter space. On the architectural side, replacing RealNVP layers with neural ordinary differential equation models [5, 21] could improve the approximation power. Additionally, the incorporation of the symmetry-respecting equivariance neural networks [9, 10, 30, 49] could further enhance the physical consistency of the learned RG transformations. With the above improvements, we envision that the RGFlow method would be a helpful tool to compute the RG transformation in high-dimensional and strongly interacting field theories.

Note added. By the time this paper was finalized, we noticed a small error in the gradient back-propagation process. This error was caused by the computation of the loss function (10). During actual training, the expectation over $P_z[z]$ was approximated by the mean of (10) over configurations sampled from the distribution. However, since the samples were detached from the distribution, the gradients of r and u could not be passed correctly to themselves. This could introduce instability in training and lead to false local minima. Fortunately, our 2D numerical results are not substantially affected. As demonstrated in figure 7, both the PyTorch gradient and the correct gradient share the same local minimum and have highly similar values. However, the behavior of PyTorch gradients for models in other dimensions remains undetermined.

The gradient problem can be fixed by the so-called policy gradient trick, which multiplies the term inside the expectation by $1 - \log P_z[z] + \log P_z[z]$. The loss function (10) then becomes

$$L(\mathbf{K}_{\text{IR}}, \theta) = \frac{1}{N} \sum_{z \sim P_z[z]} \left\| \left[\frac{\delta(\mathcal{S}_z[z; \mathbf{K}_{\text{IR}}] + \log \det J_{T^{-1}}(z))}{\delta z} - \frac{\delta \mathcal{S}_{\text{UV}}[T_\theta^{-1}(z); \mathbf{K}_{\text{UV}}]}{\delta z} \right] \right\|_2^2 \times (1 - \log P_z[z] + \log P_z[z]). \quad (32)$$

Clearly, the loss values remain unchanged. During back-propagation, the first $-\log P_z[z]$ in the parentheses is eliminated and does not pass gradients. It is then straightforward to verify that the new loss (32) provides the correct gradient even if the same approximation is applied to the expectation. However, since calculating $\log P_z[z]$ involves evaluating the ϕ^4 partition function, this new training method faces a significantly increased computational overhead and is beyond the scope of this work. We therefore would like to explore its effect in the follow-up studies.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Appendix A. Mutual information and the correlation function

In this section, we prove that the mutual information provides an upper bound to the correlation function. Given two random variables X and Y , we start with Pinsker's inequality for mutual information

$$\begin{aligned} I(X, Y) &= D_{\text{KL}}(P_{XY}(x, y) \| P_X(x) P_Y(y)) \\ &\geq \frac{1}{2} \left(\sum_{x \in X, y \in Y} |P_{XY}(x, y) - P_X(x) P_Y(y)| \right)^2. \end{aligned} \quad (33)$$

Suppose $f(x)$ and $g(y)$ are bounded functions on X and Y respectively, then the following inequality is valid:

$$0 \leq \frac{|f(x)g(y)|}{\|f(x)\|_{\infty} \|g(y)\|_{\infty}} \leq 1, \quad (34)$$

where $\|\cdot\|_{\infty}$ denotes the supremum norm. We add this fraction as a factor on the right-hand side of equation (33) to arrive at

$$\begin{aligned} I(X, Y) &\geq \frac{1}{2} \left(\sum_{x \in X, y \in Y} \left| [P_{XY}(x, y) - P_X(x) P_Y(y)] \frac{f(x)g(y)}{\|f(x)\|_{\infty} \|g(y)\|_{\infty}} \right| \right)^2 \\ &\geq \frac{1}{2} \left(\frac{\langle f(x)g(y) \rangle - \langle f(x) \rangle \langle g(y) \rangle}{\|f(x)\|_{\infty} \|g(y)\|_{\infty}} \right)^2. \end{aligned} \quad (35)$$

This is the desired relation between the mutual information and the normalized correlation function of variables f and g .

Appendix B. The linear RG loss function

For a linear RG transformation T_{θ} and the Gaussian actions (11) and (13), the loss function L in equation (10) can be simplified to

$$\begin{aligned} L &= \mathbb{E}_{z \sim P[z]} \left\| \frac{1}{2} \frac{\delta}{\delta z} \left(z^T S_z z - z^T (T_{\theta}^{-1})^T S_{UV} T_{\theta}^{-1} z \right) \right\|_2^2 \\ &= \mathbb{E}_{z \sim P[z]} \left\| z^T \left(S_z - (T_{\theta}^{-1})^T S_{UV} T_{\theta}^{-1} \right) \right\|_2^2 \\ &= \text{Tr} \left[\left(S_z - (T_{\theta}^{-1})^T S_{UV} T_{\theta}^{-1} \right)^2 S_z^{-1} \right]. \end{aligned} \quad (36)$$

The above loss function reaches its global minimum (equal to zero) if and only if $S_{UV} = T_{\theta}^T S_z T_{\theta}$ or equivalently if equation (15) holds after a Fourier transformation.

In the 1D case, our latent configuration z has alternating pattern:

$$z = (\psi_0, \xi_0, \dots, \psi_{N-1}, \xi_{N-1}). \quad (37)$$

The convolution matrix T_{θ} for the RG transformation $z = T_{\theta}(\phi)$ specified by the kernels (16) has the following block circulant form:

$$T_{\theta} = \text{bcirc} \left[\begin{pmatrix} a & c \\ 0 & f \end{pmatrix}, \begin{pmatrix} d & 0 \\ g & h \end{pmatrix}, \dots \right] \quad (38)$$

(only the nonzero blocks are shown). Projecting this T_{θ} onto the basis of Fourier harmonics, we obtain equation (17). Writing it for each of the four matrix elements, we arrive at four equations specifying the

six kernel parameters a through h :

$$\begin{aligned} (a + e^{-iq_n}d)(a + e^{iq_n}d)(r_{\text{IR}} + 2 - 2\cos q_n) + g^2 - r_{\text{UV}} - 2 &= 0, \\ (ac + e^{-iq_n}cd)(r_{\text{IR}} + 2 - 2\cos q_n) + e^{-iq_n}(fg + 1) + gh + 1 &= 0, \\ (ac + e^{iq_n}cd)(r_{\text{IR}} + 2 - 2\cos q_n) + e^{iq_n}(fg + 1) + gh + 1 &= 0, \\ c^2(r_{\text{IR}} + 2 - 2\cos q_n) + 2fh\cos q_n + f^2 + h^2 - r_{\text{UV}} - 2 &= 0. \end{aligned} \quad (39)$$

Since the above equations hold at all momenta q_n , the coefficients in front of each power of e^{iq_n} must sum up to zero. The resulting set of equations has the unique solution reported in the main text.

ORCID iDs

Yueqi Zhao  0000-0002-7751-0602

Michael M Fogler  0000-0001-8239-7221

Yi-Zhuang You  0000-0003-4080-5340

References

- [1] Asit K D, Harindranath A, Maiti J and Sinha T 2005 Investigations in $1 + 1$ dimensional lattice ϕ^4 theory *Phys. Rev. D* **72** 094503
- [2] Bachtis D, Aarts G, Di Renzo F and Lucini B 2022 Inverse renormalization group in quantum field theory *Phys. Rev. Lett.* **128** 081603
- [3] Borodinov N, Neumayer S, Kalinin S V, Ovchinnikova O S, Vasudevan R K and Jesse S 2019 Deep neural networks for understanding noisy data applied to physical property extraction in scanning probe microscopy *npj Comput. Mater.* **5** 25
- [4] Cardy J 1996 *Scaling and Renormalization in Statistical Physics (Cambridge Lecture Notes in Physics)* (Cambridge University Press)
- [5] Chen R T Q, Rubanova Y, Bettencourt J and Duvenaud D 2018 Neural ordinary differential equations *Advances in Neural Information Processing Systems (NeurIPS 2018)* vol 31 pp 6572–83
- [6] Chen X et al 2021 Hybrid machine learning for scanning near-field optical spectroscopy *ACS Photonics* **8** 2987–96
- [7] Chen Y, Lu L, Karniadakis G E and Negro L D 2020 Physics-informed neural networks for inverse problems in nano-optics and metamaterials *Opt. Express* **28** 11618–33
- [8] Chung J-H and Kao Y-J 2021 Neural Monte Carlo renormalization group *Phys. Rev. Res.* **3** 023230
- [9] Cohen T, Geiger M and Weiler M 2020 A general theory of equivariant CNNs on homogeneous spaces *Advances in Neural Information Processing Systems (NeurIPS 2019)* vol 32 pp 9113–24
- [10] Cohen T S and Welling M 2016 Group equivariant convolutional networks *Proc. 33rd Int. Conf. on Machine Learning (ICML)* vol 48 pp 2990–9 (available at: <https://proceedings.mlr.press/v48/cohen16.html>)
- [11] Coleman T F and Yuying Li 1994 On the convergence of interior-reflective Newton methods for nonlinear minimization subject to bounds *Math. Program.* **67** 189–224
- [12] Coleman T F and Li Y 1996 An interior trust region approach for nonlinear minimization subject to bounds *SIAM J. Optim.* **6** 418–45
- [13] Delcamp C and Tilloy A 2020 Computing the renormalization group flow of two-dimensional ϕ^4 theory with tensor networks *Phys. Rev. Res.* **2** 033278
- [14] Di Sante D, Medvidović M, Toschi A, Sangiovanni G, Franchini C, Sengupta A M and Millis A J 2022 Deep learning the functional renormalization group *Phys. Rev. Lett.* **129** 136402
- [15] Dinh L, Sohl-Dickstein J and Bengio S 2017 Density estimation using Real NVP *Proc. 5th Int. Conf. on Learning Representations (ICLR 2017)* (<https://doi.org/10.48550/arXiv.1605.08803>)
- [16] Dong Z, Cao W, Zhang M, Tao D, Chen Y and Zhang X 2024 CktGNN: circuit graph neural network for electronic design automation
- [17] Fisher M E 1974 The renormalization group in the theory of critical behavior *Rev. Mod. Phys.* **46** 597–616
- [18] Giataganas D, Huang C-Y and Lin F-Li 2022 Neural network flows of low q -state Potts and clock models *New J. Phys.* **24** 043040
- [19] Gordon O, D'Hondt P, Knijff L, Freeney S E, Junqueira F, Moriarty P and Swart I 2019 Scanning tunneling state recognition with multi-class neural network ensembles *Rev. Sci. Instrum.* **90** 103704
- [20] Gordon O M, Hodgkinson J E A, Farley S M, Hunsicker E L and Moriarty P J 2020 Automated searching and identification of self-organized nanostructures *Nano Lett.* **20** 7688–93
- [21] Grathwohl W, Chen R T Q, Bettencourt J, Sutskever I and Duvenaud D 2018 FFJORD: free-form continuous dynamics for scalable reversible generative models *Proc. 7th Int. Conf. on Learning Representations (ICLR 2019)* (<https://doi.org/10.48550/arXiv.1810.01367>)
- [22] Hammond A M and Camacho R M 2019 Designing integrated photonic devices using artificial neural networks *Opt. Express* **27** 29620–38
- [23] Hou W and You Y-Z 2023 Machine learning renormalization group for statistical physics *Mach. Learn.: Sci. Technol.* **4** 045010
- [24] Hong-Ye H, Shuo-Hui Li, Wang L and You Y-Z 2020 Machine learning holographic mapping by neural network renormalization group *Phys. Rev. Res.* **2** 023369
- [25] Kadanoff L P 1976 Notes on Migdal's recursion formulas *Ann. Phys.* **100** 359–94
- [26] Kadanoff L P and Houghton A 1975 Numerical evaluations of the critical properties of the two-dimensional ising model *Phys. Rev. B* **11** 377–86
- [27] Kingma D and Jimmy B 2015 Adam: a method for stochastic optimization *Int. Conf. on Learning Representations (ICLR)* (San Diego, CA, USA)
- [28] Kobayev I, Prince S J D and Brubaker M A 2021 Normalizing flows: an introduction and review of current methods *IEEE Trans. Pattern Anal. Mach. Intell.* **43** 3964–79
- [29] Koch-Janusz M and Ringel Z 2018 Mutual information, neural networks and the renormalization group *Nat. Phys.* **14** 578–82

- [30] Kondor R and Trivedi S 2018 On the generalization of equivariance and convolution in neural networks to the action of compact groups *Proc. 35th Int. Conf. on Machine Learning (ICML)* vol 80 pp 2747–55 (available at: <https://proceedings.mlr.press/v80/kondor18a.html>)
- [31] Lee C-H et al 2020 Deep learning enabled strain mapping of single-atom defects in two-dimensional transition metal dichalcogenides with sub-picometer precision *Nano Lett.* **20** 3369–77
- [32] Lee C H and Xiao-Liang Q 2016 Exact holographic mapping in free fermion systems *Phys. Rev. B* **93** 035112
- [33] Lenggenhager P M, Gokmen D E, Ringel Z, Huber S D and Koch-Janusz M 2020 Optimal renormalization group transformation from information theory *Phys. Rev. X* **10** 011037
- [34] Loinaz W and Willey R S 1998 Monte Carlo simulation calculation of the critical coupling constant for two-dimensional continuum ϕ^4 theory *Phys. Rev. D* **58** 076003
- [35] Migdal A A 1975 Phase transformations in gauge and spin lattice systems *JETP* **42** 743
- [36] Migdal A A 1975 Recursion equations in gauge field theories *JETP* **42** 413
- [37] Milsted A, Haegeman J and Osborne T J 2013 Matrix product states and variational methods applied to critical quantum field theory *Phys. Rev. D* **88** 085030
- [38] Papamakarios G, Nalisnick E, Rezende D J, Mohamed S and Lakshminarayanan B 2021 Normalizing flows for probabilistic modeling and inference *J. Mach. Learn. Res.* **22** 2617–57:2680
- [39] Rezende D J and Mohamed S 2015 Variational inference with normalizing flows *Proc. 32nd Int. Conf. on Machine Learning - Volume 37, ICML' (Lille, France)* vol 15 (JMLR.org) pp 1530–8
- [40] Ron D, Brandt A and Swendsen R H 2021 Monte Carlo renormalization-group calculation for the $d = 3$ Ising model using a modified transformation *Phys. Rev. E* **104** 025311
- [41] Rychkov S and Vitale L G 2015 Hamiltonian truncation study of the ϕ^4 theory in two dimensions *Phys. Rev. D* **91** 085011
- [42] Schaich D and Loinaz W 2009 Improved lattice measurement of the critical coupling in ϕ^4_2 theory *Phys. Rev. D* **79** 056008
- [43] Serone M, Spada G and Villadoro G 2018 $\Lambda\phi^4$ theory—part I. The symmetric phase beyond NNNNNNNNLO *J. High Energy Phys.* **JHEP08(2018)148**
- [44] Shankar R 2017 *Quantum Field Theory and Condensed Matter: An Introduction* (Cambridge University Press)
- [45] Sheshmani A, You Y-Z, Wenbo F and Azizi A 2023 Categorical representation learning and RG flow operators for algorithmic classifiers *Mach. Learn.: Sci. Technol.* **4** 015012
- [46] Li S-H and Wang L 2018 Neural network renormalization group *Phys. Rev. Lett.* **121** 260601
- [47] Sugihara T 2004 Density matrix renormalization group in a two-dimensional $\lambda\phi^4$ hamiltonian lattice model *J. High Energy Phys.* **JHEP05(2004)007**
- [48] Ueda A and Oshikawa M 2023 Finite-size and finite bond dimension effects of tensor network renormalization *Phys. Rev. B* **108** 024413
- [49] Weiler M, Geiger M, Welling M, Boomsma W and Cohen T 2018 3D steerable CNNs: Learning rotationally equivariant features in volumetric data *Advances in Neural Information Processing Systems (NeurIPS 2018)* vol 31 (Learning Rotationally Equivariant Features in Volumetric Data) (<https://doi.org/10.48550/arXiv.1807.02547>)
- [50] Wilson K G 1975 The renormalization group: critical phenomena and the kondo problem *Rev. Mod. Phys.* **47** 773–840
- [51] Wilson K G and Kogut J 1974 The renormalization group and the ϵ expansion *Phys. Rep.* **12** 75–199
- [52] Qi X-L 2013 Exact holographic mapping and emergent space-time geometry (arXiv:1309.6282)
- [53] Yingfei G, Lee C H, Wen X, Cho G Y, Ryu S and Xiao-Liang Q 2016 Holographic duality between $(2 + 1)$ -dimensional quantum anomalous Hall state and $(3 + 1)$ -dimensional topological insulators *Phys. Rev. B* **94** 125107
- [54] Zhao Y, Chen X, Yao Z, Liu M K and Fogler M M 2023 Deep-learning-aided extraction of optical constants in scanning near-field optical microscopy *J. Appl. Phys.* **133** 133105
- [55] Ziatdinov M, Dyck O, Maksov A, Xufan Li, Sang X, Xiao K, Unocic R R, Vasudevan R, Jesse S and Kalinin S V 2017 Deep learning of atomically resolved scanning transmission electron microscopy images: chemical identification and tracking local transformations *ACS Nano* **11** 12742–52