Deep Learning Cosmic Ray Transport from Density Maps of Simulated, Turbulent Gas

Chad Bustard[1], **John Wu**[2,3]

[1] Kavli Institute for Theoretical Physics, University of California - Santa Barbara, Kohn Hall, Santa Barbara, CA 93107, USA

[2] Space Telescope Science Institute, 3700 San Martin Dr, Baltimore, MD 21218

[3] Department of Physics & Astronomy, Johns Hopkins University, 3400 N Charles St, Baltimore, MD 21218

E-mail: bustard@ucsb.edu

Abstract. The coarse-grained propagation of Galactic cosmic rays (CRs) is traditionally constrained by phenomenological models of Milky Way CR propagation fit to a variety of direct and indirect observables; however, constraining the fine-grained transport of CRs along individual magnetic field lines – for instance, diffusive vs streaming transport models – is an unsolved challenge. Leveraging a recent training set of magnetohydrodynamic turbulent box simulations, with CRs spanning a range of transport parameters, we use convolutional neural networks (CNNs) trained solely on gas density maps to classify CR transport regimes. We find that even relatively simple CNNs can quite effectively classify density slices to corresponding CR transport parameters, distinguishing between streaming and diffusive transport, as well as magnitude of diffusivity, with class accuracies between 92% and 99%. As we show, the transport-dependent imprints that CRs leave on the gas are not all tied to the resulting density power spectra: classification accuracies are still high even when image spectra are flattened (85% to 98% accuracy), highlighting CR transport-dependent changes to turbulent phase information. We interpret our results with saliency maps and image modifications, and we discuss physical insights and future applications.

1. Introduction

1.1. Cosmic Ray Fundamentals

Galaxies are complex, dynamic systems with collisional components such as gas reservoirs, and collisionless components that primarily interact through gravity (such as stars and dark matter). In a broad sense, the collisional composition of galaxies can be divided as follows: there is non-relativistic, typically ionized gas that we are most accustomed to thinking about, there are cosmic rays (CRs), which are high-energy, charged particles that travel through the Universe at close to the speed of light, and there are magnetic fields, which couple to both non-relativistic gas and relativistic CRs through electromagnetic forces. Each of these components shapes the gas flows that

regulate star formation and the long-term evolution of galactic ecosystems, but there are significant unknowns with each component and their interplay.

In this paper, we concern ourselves with how CRs, on large scales†, transfer momentum and energy with the surrounding non-relativistic gas, which is very dependent on the highly uncertain and scale-dependent motion of the CR fluid relative to the background gas. On very large scales, we can average over many of the smaller, turbulent fluctuations in the Universe, and therefore average over the tangled magnetic field lines that guide and scatter CRs. This large-scale, *coarse-grained* CR propagation is what is constrained by current state-of-the-art phenomenological models [1]. These models make informed assumptions on the geometry of the Milky Way, inject CRs from their likely sources within the Milky Way disk, propagate CRs according to some plausible paradigms with tune-able parameters, and calculate a variety of direct and indirect CR indicators: for instance, gamma-ray emission from interactions between hadrons and CR protons, radio synchrotron emission from spiraling CR electrons, and secondary products of spallation, the direct collision of CRs with other gas particles in the Universe. The best configuration, which minimizes the differences between the model output and real observations, is one in which the coarse-grained transport of CRs is diffusive and energy-dependent [2, 1]; however, these models cannot tell us the zoomed-in, *fine-grained* CR transport along individual field lines.

The fine-grained transport depends on the source and type of hydromagnetic waves that scatter and confine CRs (see e.g. [3, 4] for recent reviews). If CRs scatter off compressible fast modes [5] that are created by external turbulence and cascade down to the CR gyroscale (≈ 0.1 AU for a GeV CR in the Milky Way), then fine-grained CR transport is believed to be diffusive and predominantly parallel to the local magnetic field. On the other hand, if the scattering waves are created by the CRs themselves through the so-called "streaming instability" [6, 7], then CR transport is referred to as "streaming", which can be a mixture of field-aligned diffusion and additional field-aligned advection at the Alfvén speed $v_A = B/\sqrt{4\pi\rho}$, where B is the magnetic field strength and ρ is the gas density.

For a multitude of reasons, identifying the true, fine-grained CR transport mode is crucial. Despite representing only a billionth of all particles in the Milky Way, CRs on a whole have as much energy as normal, non-relativistic gas [8], and it is clear from a veritable explosion of work in the last decade (e.g. [9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 4]) that the content of CRs in various astrophysical environments and the dynamical and thermodynamical influence of CRs on the surrounding gas sensitively depend on this transport. For example, in simulations of the Large Magellanic Cloud (LMC), a neighboring satellite galaxy of the Milky Way, if one allows CRs to stream, the galaxy remains largely intact over long periods of time as CRs easily lose pressure and escape the galaxy; however, if one replaces streaming with a small diffusivity instead, CRs build up a large pressure gradient in the galaxy and expel gas in a large-scale "galactic

†On "large" scales, typically greater than a parsec, CRs can be collectively described as a relativistic fluid instead of individual particles.

wind" [19]. To reconcile observed galaxy gas contents with simulations and to predict the future gas content of galaxies, including those like the LMC that will eventually collide with the Milky Way, we need to know the true mixture of streaming vs diffusive CR transport.

So how can we determine whether CR transport is streaming or diffusive? One intriguing approach, motivated by very recent results, is to use the distinct, transport-dependent imprints that CRs leave on their surroundings. The basis for this idea is that diffusing and streaming CRs interact with gas fluctuations in fundamentally different ways. Denoting the CR pressure as $P_{\rm CR}$, diffusing CRs have flux $F_{\rm CR} \propto \nabla P_{\rm CR}$, which introduces a CR perturbed force that is proportional to velocity and creates a $\pi/2$ phase shift between CR pressure and gas density perturbations. Akin to a damped harmonic oscillator, this damps the waves [20], leading to CR acceleration. Streaming CRs, with flux $F_{\rm CR} \propto P_{\rm CR}$, do not induce such a phase shift and have decreased acceleration rates [21], but they transfer energy to the gas and can drive unique instabilities, for instance of acoustic waves in highly magnetized plasmas [22] as seen recently in idealized 1D simulations [23, 24].

Transport-dependent impacts on gas are also readily apparent in fully 3D simulations. Bustard and Oh 2023 [25] simulated CR-gas interactions in subsonic, compressive turbulence and found that turbulent energy spectra change dramatically depending on CR transport mode, with all other variables (CR pressure, stirring rate, etc.) held fixed. Namely, when CR diffusion dominates, CRs take energy from the gas and gain energy themselves‡, introducing cut-offs and new slopes to kinetic energy spectra compared to a no-CR case. CR streaming alters this damping [21], affecting turbulent spectra, gas thermodynamics, and density structures in a distinctly different manner [25].

1.2. Goals of this Work

Overall, the Bustard and Oh 2023 simulation suite provides terabytes of unstructured gas density images stemming from otherwise identical simulations but with different CR transport assumptions. The primary goal of this paper is to explore whether deep convolutional neural networks (CNNs) can learn to accurately predict the CR transport encoded in each image, and more importantly, whether subsequent network interpretation using image manipulation and saliency maps can help illuminate the most salient, distinguishing features of gas density maps. To that end, our aim is to train a CNN to high enough accuracy to enable useful interpretation and reveal new insights into how CRs affect their surroundings. In the following exploratory analysis, we use density slices from the Bustard and Oh 2023 simulation suite as our training and validation data, and we train and fine-tune CNNs using PyTorch [28], a popular and open-source Python-based deep learning framework, to classify density images into one of five sets of simulations, varying only in the CR transport model assumed.

‡The subsequent CR energy gain is known as turbulent reacceleration [26, 27].

A second question, which we largely defer to future work, is whether these neural networks trained on simulations can be used accurately in production with real observations as inputs. This possibility is quite interesting: instead of requiring the full, and expensive to acquire, multi-wavelength observations input into phenomenological models [29, 1], all one theoretically needs are images of HI (neutral hydrogen) density obtained from a high-resolution survey. Given the highly idealized nature of even these state-of-the-art turbulence simulations and the significant uncertainty as to whether idealized simulations capture the density map differences of real observations§, it is premature to conduct a full study of this possibility. Instead, we discuss this domain adaptation in Section 5 and briefly show that, in light of our results in Section 3, a universal challenge complicating all astronomical analyses is especially relevant here: the depth of 3D structures that forms a 2D image is highly uncertain, and varying this depth significantly changes the accuracy of our network.

The outline of this paper is as follows. In Section 2, we recap the simulations of Bustard and Oh 2023, describe the data preprocessing steps we take, and outline the basic components of our CNN architecture. In Section 3, we present our classification results, first on the entire fiducial dataset spanning all five classes. We also present our interpretation of these results using saliency maps, and we probe the limitations of CNNs further by flattening the power spectra of our images and Gaussian filtering our input images. We conclude in Section 5. In 4.2, we explore how well a network trained on single-cell-thick slices of a density cube can classify more realistic projections over multiple cells, highlighting the need for additional training on a larger, realistic, and more diverse image set.

Code availability: All Python scripts used in this work are hosted at `https://github.com/bustardchad/ML_Turb`, including descriptive Jupyter notebooks.

Data availability: A subset of data is hosted through the Harvard Dataverse at `https://doi.org/10.7910/DVN/WBY5CX`

2. Training Data and CNN Architecture

2.1. Simulation Sets and Labels

The data for this project comes from the Bustard and Oh 2023 turbulent box simulation suite, and we encourage readers to see Section 2 of [25] for further details. As a brief recap, these simulations are all run using the Athena++ magnetohydrodynamics (MHD) code [30] with an additional module that includes CRs as a relativistic fluid with adiabatic index $\gamma_c = 4/3$ and with energy and flux coupled to the normal MHD equations (see [31] for more details), including terms for field-aligned CR streaming and diffusion. The non-relativistic gas is treated as an isothermal (constant temperature) fluid with adiabatic index $\gamma_g = 1$. Purely compressive turbulence is stirred according to

§For example, simulations may not be sufficiently converged with resolution, or other potentially dominant complications due to dust or multiphase gas can exist.

Class Name	CR Transport Parameters
`MHD`	No CRs
`CR_Advect`	$\kappa \sim 0$
`CR_Diff_Fiducial`	$\kappa = \kappa_f \sim 0.1 L_0 c_s$
`CR_Diff100`	$\kappa = 100\kappa_f \sim 10 L_0 c_s$
`CR_withStreaming`	$\kappa = \kappa_f$ + streaming

Table 1. Class labels and image set names. For each simulation, the following are constant: $\beta = P_g/P_B \sim 10$, $\eta = P_g/P_{CR} \sim 1$ (except $\eta = 0$ for the `MHD` class), the box size $(2L)^3$, the outer eddy size $L_0 \sim 2L/3$, and all stirring parameters (see [25]).

an Ornstein-Uhlenbeck random process [32] centered on scale L in a cubic box of size $(2L)^3$, leading to an effective turbulent outer scale of $L_0 \approx 2L/3$. In each simulation, the turbulent stirring rate, correlation time, etc. are all kept fixed, and in this study, we focus on the simulations that, in absence of CRs, produce a sonic Mach number $M_s \sim 0.15$, defined as the ratio of the turbulent velocity v to the gas sound speed c_s. The initial composite mixture of gas, magnetic fields, and CRs is constant for all simulations, and has initial gas-to-magnetic pressure ratio $\beta = P_g/P_B \sim 10$ and CR-to-gas pressure ratio $\eta = P_{CR}/P_g \sim 1$, except the MHD-only simulation, for which there are no CRs. The initial magnetic field configuration is in the $\hat{x}$-direction. The stirring generates sub- to trans-Alfvénic ($\mathcal{M}_A = v/v_A < 1$) turbulence, and with purely compressive forcing (rather than solenoidal forcing), there is no appreciable amplification of the magnetic field.

Importantly, compressive motions transfer kinetic energy to the CRs. Prior analyses showed that the rate of this CR energization depends on CR transport mode and the gas-to-magnetic pressure ratio β[21], while Bustard and Oh 2023 showed that this transport-dependent energy transfer affects the turbulent kinetic energy cascade and the spectra of gas density structures. Namely, there is a "sweet-spot" CR diffusion coefficient $\kappa \sim 0.1 L_0 c_s$ where CRs most severely damp turbulent fluctuations, leading to a steeper spectral slope and a lack of small-scale power in the cascade compared to simulations with non-optimal CR diffusivity. Our simulation classes are shown in Table 1, with expanded descriptions of each class given below:

- `MHD`, a no-CR ($\eta = P_g/P_{CR} = 0$) control case. Turbulence in this case is entirely formed by MHD effects. On well-resolved scales (wave numbers $\mathrm{k} < 10 - 20$), there is significant power, as CRs are not present to play a damping role.
- `CR_Advect`, with roughly equal pressure contributions from CRs and gas ($\eta = P_g/P_{CR} \sim 1$), but with no CR diffusivity ($\kappa = 0$). We refer to this case as `CR_Advect` because CRs only *advect* with the gas. The major physical difference, then, is that the composite CR and gas mixture has an effective equation of state somewhere between that of the isothermal, non-relativistic gas, where pressure and density are related by $P \sim \rho$, and a relativistic gas, where $P \sim \rho^{4/3}$. This slightly affects the compressibility of the gas and the resulting density images, but as we see from

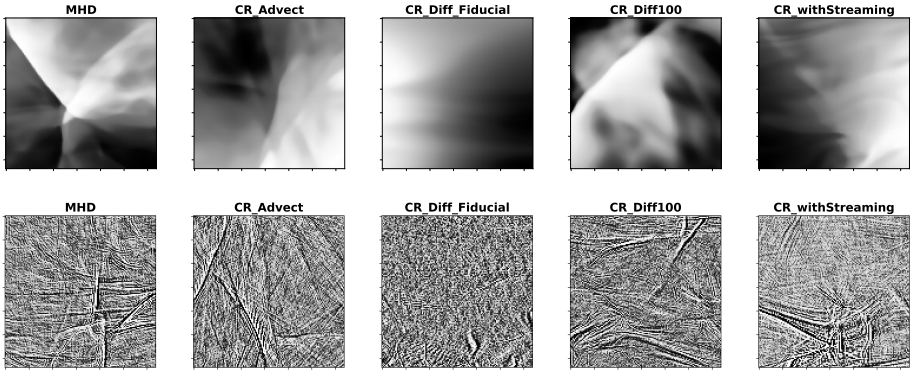

Figure 1. Example gas density images for each class. The top row shows unaltered images from the `Full Power` set, while the bottom row shows another, distinct set of images after their power spectra have been flattened (from the `Flattened Power` set). Note in the top row some of the major differences between `CR_Diff_Fiducial`, which shows very smooth transitions between over- and under-dense gas, and e.g. `MHD`, which has sharp transitions and more small-scale structure.

Figure 2, the 1D density power spectrum is very similar to the `MHD` case, suggesting these classes will be hard to disentangle.

- `CR_Diff_Fiducial`, where CRs are present with a fiducial diffusivity $\kappa_f \sim 0.1 L_0 c_s$. This diffusivity optimizes the energy transfer between gas fluctuations and CRs, leading to the most significant damping. From Figure 2, this decreases the power in intermediate wavenumber fluctuations.
- `CR_Diff100`, where $\kappa \rightarrow 100\kappa_f$. With such a high diffusivity, CRs flow over gas fluctuations so quickly that they don't damp them as effectively. This leads to gas density power spectra intermediate between the `MHD` and `CR_Diff_Fiducial` cases.
- `CR_withStreaming`, which includes both fiducial diffusion and CR streaming. This case is of particular interest because of the unique changes that streaming imparts on the turbulence. Instead of CRs taking energy from gas motions and keeping it, the amount of turbulent damping is lower, and much of the energy that CRs receive is deposited back into the gas as heat at scales far larger than the typical dissipation scale. The resulting turbulent energy spectrum does not display such an obvious cut-off or change in spectral slope, but is instead suppressed almost uniformly across all scales (see Figure 8 in [25]). If one scales and normalizes the resulting density images, as we do in this work, the streaming spectrum is almost exactly the same as the `MHD`, `CR_Advect`, and `CR_Diff100` spectra, as we see in Figure 2.

2.2. Training Data and Preprocessing

Each simulation snapshot contains 512^3 cells, and for each simulation class, we utilize 6 time snapshots temporally separated by at least an (outer-scale) eddy turnover time

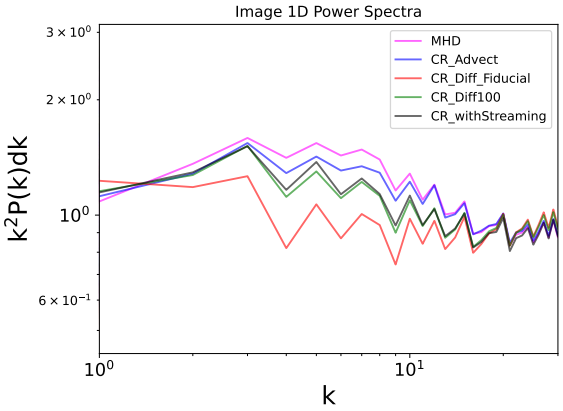

Figure 2. Average 1D gas density power spectra (further multiplied by k^2) over a batch of 512 images from the test set, showing noticeable spectral differences between `CR_Diff_Fiducial` and other sets but relatively small differences between the other CR sets.

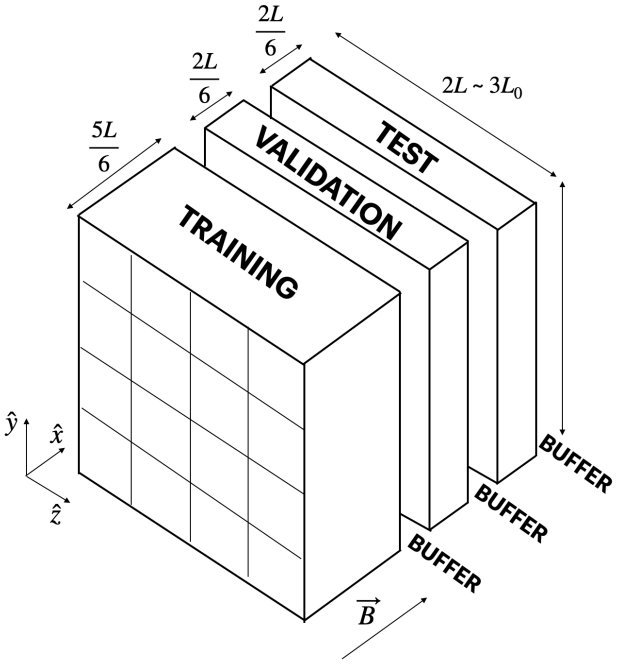

Figure 3. Cartoon showing how each data snapshot with total volume of $(2L)^3$ is split into training, validation, and test sets. Our training set occupies a volume of $(2L \times 2L \times 5L/6)$, while our validation and test sets are 2.5 times smaller with volumes of $(2L \times 2L \times 2L/6)$. Each set is spatially separated from other sets by a buffer of width $L/6$ in the $\hat{x}$ direction to decrease correlations. Within each set, images of size 128×128 cells are created by slicing in the y-z plane, perpendicular to the initial magnetic field $\vec{B}$.

to ensure snapshots are sufficiently uncorrelated. Our process for splitting these data cubes into images largely follows that of [33], who similarly analyzed MHD turbulence simulations to classify sub-Alfvénic vs super-Alfvénic regimes. To create our image sets, we cut the box into 512 one-cell-thick slices of dimension 512×512 in the y-z plane (transverse to the initial guide magnetic field), and from each of those slices, we create 16 images of size 128×128.

Since eddies in magnetized turbulence are elongated along the background magnetic field direction $\hat{x}$ [34], slices in the x-z or x-y planes will contain imprints of the original magnetic field. The extent of this eddy anisotropy, which imprints on gas density images, would not bias our analysis because it is indeed a physical outcome of our simulations that start from exactly the same initial conditions. Nevertheless, to ensure that this anisotropy, which is dominantly caused by magnetic field effects rather than CR effects, is not a feature that our network can use to distinguish different CR transport modes, we follow [33] and slice across the magnetic field axis, forcing the network to find other distinguishing image characteristics more likely to be caused by CRs.

We also note here that, by taking a one cell thick slice, we are effectively integrating over structures on the scale of a simulation cell width. How well this reflects the projection depth of a real astronomical image is complicated and very dependent on the astrophysical environment, as we discuss in Section 4.2. In this paper, however, we focus on how neural network interpretation can help us derive new insights from simulations, and our slicing choice is sufficient.

From here, we must be careful to spatially separate the training, validation, and test slices so that structures correlated in the $\hat{x}$-direction do not bleed between sets and introduce correlations. To decrease the chances of this, we put spatial buffers between the training, validation, and test sets. The training set occupies a width of $5L/6$, followed by a buffer of width $L/6$, then a validation set of width $2L/6$, then a buffer, then a test set of width $2L/6$, then a final buffer (Figure 3). Our results do not seem particularly sensitive to buffer size, except in the case with *no* buffer where our saliency maps (Section 3.2) were dominated by pixel-scale regions, indicative of the CNN "memorizing" regions of the training set that were correlated with the validation set. This problem was particularly evident when we created training and validation sets by randomly choosing slices from the 3D data volumes; in this case, structures very much span across images from both sets, leading to a network with no ability to generalize to unseen data. An alternative way to split training, validation, and test sets could be to separate them temporally. For instance, training data could comprise snapshots 1-4, validation snapshot 5, and test snapshot 6; however, this means images within each set are not well-separated in time. By instead creating sets that span across all times available, our network is trained, validated, and tested on more diverse manifestations of the turbulent gas-CR interactions.

Within each set, we fiducially keep half of the images; we primarily do this to keep our dataset sizes small enough to be loaded into RAM, but this can also help decrease spatial correlations within each set. To further decrease correlations, we randomly flip

the images both horizontally and vertically, each with a 50% probability. In all, our fiducial training, validation, and test sets contain $\sim$ 10,000, 4,000, and 4,000 images per class (50,000, 20,000, and 20,000 total). Each image is then preprocessed as follows:

(i) The density is logarithmically scaled. Because turbulent density probability distribution functions (PDFs) are roughly lognormal, this scaling brings out more features that would otherwise be sub-dominant compared to the most dense regions.

(ii) The images are histogram equalized using the exposure method from scikit-image [35] such that image pixels have a roughly equal distribution of values from 0 to 1. As noted in [33], this step is a common preprocessing step used to optimize the CNN, but it eliminates density PDF information from our images. With one of our goals to see if neural networks can find information *beyond* PDF and spectral information, this is perfectly acceptable.

Images processed in this way encompass our `Full Power` image set in that they retain spectral information. As in [33], we also create a `Flattened Power` image set with no spectral information by applying a fast Fourier transform to each image and setting the Fourier power to unity; this happens in between steps (i) and (ii) above. In this case, the neural network is left to only distinguish image classes based on image *phase* information.

2.3. Neural Network Details

Convolutional neural networks (CNNs) are a powerful deep learning architecture for computer vision, and as they have now been employed for various tasks in astrophysics, we do not give a long introduction to them here. Instead, we describe the key components and our choices for number of layers, number of trainable parameters, etc., and refer the audience to a recent review of deep learning in astrophysics (i.e. [36]).

The building blocks of CNNs are convolution layers, pooling layers, and fully connected layers, followed in this classification application by a softmax output layer that generates a probability of the input image belonging to each class. For the results presented here, we use 4 convolutional layers, each followed by batch normalization and SiLU activations. These layers take an input array (in the first layer, this input is the 128×128 image), and apply filters to sub-patches of the input, thereby generating many convolutions of the input. Batch normalization then normalizes the layers' inputs by re-centering and re-scaling them, making training faster and more stable.

After these 4 convolution layers, we apply a pooling layer and then apply dropout with 25% probability. We then flatten the output before sending it to a fully connected layer, where all neurons from the previous layer are connected to all neurons of the next layer. The output of this final layer, after going through a softmax activation, is a vector of probabilities that the input image corresponds to each class. For our full model with 5 classes, this vector has a length of 5, and when we make our final prediction of which class the image came from, we choose the class with highest probability.

All-in-all, this fiducial network, containing 29,749 trainable parameters, is appropriately sized for our dataset of $\sim 50,000$ training images; adding more layers leads to overfitting (high accuracy on training data but poor generalization to unseen data), while decreasing the number of layers leads to underfitting (poorer accuracy on training data). Networks for the `Full Power` and `Flattened Power` datasets are trained for 40 and 25 epochs, respectively, beyond which the models begin to overfit.

To speed up training, we employ mini-batch gradient descent with 64 images per batch. Weights and biases are updated during training using the AdamW optimizer [37] with weight decay of 10^{-4} and a learning rate of 10^{-3} in the `Full Power` case and 5×10^{-4} in the `Flattened Power` case. This gradient descent method, which is a modification to the popular Adam method [38], decouples weight decay from the gradient update steps and improves generalization performance. The loss function that AdamW seeks to minimize is the cross-entropy loss between the predicted distribution and the true class distribution.

All of the above choices were motivated by a limited, manual hyperparameter study, where we varied the learning rate between 10^{-3} and 10^{-4}, the batch size between 8 and 256, and the dropout fraction between 0 and 0.5. We also tested the ReLU activation function instead of the SiLU activation function in hidden layers of our network, ultimately finding insignificant differences in training time and accuracy. The continuously differentiable SiLU behaves similarly to other activation functions (e.g., [39, 40, 41]), which are robust against the "dying neuron" problem with ReLU, and have been shown to improve performance in astronomical tasks (e.g., [42, 43]).

3. Results

3.1. Full Spectra Results

In Table 2, we display several commonly used metrics for multi-class machine learning problems, which depend on the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). The accuracy is defined as (TP + TN) / (TP + TN + FP + FN), the precision is defined as the TP / (TP + FP), the recall is defined as the TP / (TP + FN), and the F1 score is the harmonic mean between the purity and recall. The precision can be considered a measure of "purity" while the recall can be thought of as "completeness" for CNN predictions. Figure 4 shows the confusion matrix for the `Full Power` test set, created with scikit-learn [44] and showing raw counts of images with predicted vs true labels.

The accuracy for each class is quite high, ranging from 92.0% for `CR_Diff100` to 99.2% for `CR_Diff_Fiducial`. Precision and recall vary more significantly, leading to generally lower F1 scores. For instance, recall ranges from $\sim$ 78% for the `MHD` class to 100% for the `CR_Diff_Fiducial` class, but especially for the `MHD` class, low recall is compensated by high precision (99.5% for `MHD`). This tendency for a CNN to trade recall for precision or vice versa is a common and nonlinear behavior; therefore, it is critical

Data set	Accuracy	Precision	Recall	F1-Score
`Full Power`				
`MHD`	95.5	99.5	77.9	87.4
`CR_Advect`	95.6	82.8	98.4	89.9
`CR_Diff_Fiducial`	99.2	96.0	100.0	98.0
`CR_Diff100`	92.0	77.7	84.2	80.8
`CR_withStreaming`	94.2	89.4	80.8	84.9
`Flattened Power`				
`MHD`	88.5	99.6	42.8	59.9
`CR_Advect`	96.7	96.7	86.4	91.2
`CR_Diff_Fiducial`	97.9	99.7	89.8	94.5
`CR_Diff100`	90.2	72.5	82.4	77.1
`CR_withStreaming`	85.7	58.7	96.2	72.9

Table 2. A table of metrics comparing classification results on `Full Power` simulations and on `Flattened Power` simulations. The accuracy, precision, recall, and F1 scores are shown as percentages.

to report multiple summary statistics and combined metrics like the F1 score¶.

The average recall is 88.3%, brought down most significantly by the `MHD` case, which is confused for other classes 22% of the time. In particular, `MHD` is incorrectly labeled as `CR_Advect` 11.4% of the time and as `CR_Diff100` 8.9% of the time. These confusions make physical sense: advecting CRs do not sap any energy from turbulent fluctuations. Instead, the inclusion of CRs only changes the composite gas+CR equation of state because relativistic CRs have a $\gamma = 4/3$ adiabatic index instead of a $\gamma = 5/3$ index for a non-relativistic gas. The resulting images are quite comparable. CRs with large diffusivity (`CR_Diff100`) do not appreciably interact with fluctuations either; their fast transport (on a short diffusive timescale $\tau_{\rm diff} \sim L^2/\kappa$) means they pass over the flow too quickly for eddies to interact with the CRs during an eddy turnover time $\tau_{\rm eddy} \sim L/v$, essentially leaving turbulence and the resulting density image untouched.

One might wonder whether the MHD simulations are necessary when, for instance, we know quite certainly that MHD-only is a poor approximation in the Milky Way interstellar medium where $P_{\rm CR} \sim P_g$ [8]. To test this scenario, we also trained and fine-tuned networks on only the 4 CR classes. For brevity, we do not show the resulting confusion matrix, but we note that we obtained very similar statistics with only slightly boosted F1 scores. This implies that the full, 5-class network is capable of discriminating between CR classes, despite being presented with confusing MHD images.

¶In practice, one could change the softmax function in our network to include a tunable parameter α, i.e. softmax(α,z) = exp(-αz)/$\sum_z$exp(-αz), and evaluate the F1 score on the validation set over a grid of α values to find the optimal trade-off between precision and recall averaged over all classes. However, this does not guarantee that one will find an α that simultaneously maximizes the F1 score for each individual class, and we proceed with a default value of $\alpha = 1$.

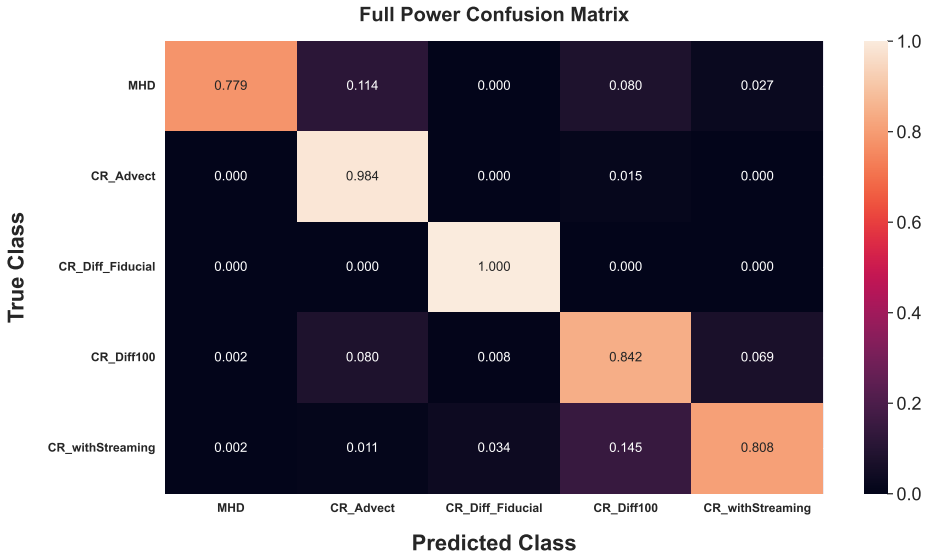

Figure 4. Confusion matrix for the `Full Power` set of test images, showing the fraction of images in each predicted class vs their true class. Fractions are obtained by taking the raw number of images in each category and dividing by 3981, the number of images per class. Class accuracies range from 92.0% for the `CR_Diff100` class to 99.2% for the `CR_Diff_Fiducial` class, which also achieves a perfect recall (see Table 2). The average recall is $\approx 88.3\%$, brought down significantly by the `MHD` images.

Most impressively, the `CR_Diff_Fiducial` and `CR_withStreaming` classes, which differ only in that CR streaming is included in addition to fiducial diffusivity, are well-distinguished, with the network achieving 94.2% accuracy on `CR_withStreaming` and only rarely (3.4% of the time) confusing `CR_withStreaming` for the `CR_Diff_Fiducial` class. Instead, `CR_withStreaming` is confused for `CR_Diff100` $\approx 14.5\%$ of the time, likely because additional CR streaming means CRs are propagating faster along field lines, somewhat akin to faster diffusion. How fast is streaming transport? In these simulations, turbulence is sub-Alfvénic, meaning the characteristic CR transport speed (the Alfvén speed v_A) is faster than the turbulent velocity‖, meaning the CR transport time $\tau_{\rm stream} \sim L/v_A < L/v \sim \tau_{\rm eddy}$, similar to the fast diffusion case where $\tau_{\rm diff} \sim L^2/\kappa < L/v \sim \tau_{\rm eddy}$. Previous work [25], however, hints that additional discriminating information will be present, such as transport-dependent ratios of compressive vs solenoidal motions in the gas, which likely accounts for some of the accurate differentiation between `CR_withStreaming` and other CR classes. In all, the accuracies we obtain with our relatively simple network are high enough to continue with network interpretation, and misclassification trends shown by our confusion matrices already reveal distinct

‖In fact, this characteristic transport speed is a lower limit only realized when CRs are well-coupled to Alfvén waves, which only occurs when CR pressure gradients are aligned with the magnetic field. In turbulence, these vectors are frequently misaligned [21], leading to macroscopic CR decoupling from waves and the so-called "bottleneck effect", where CRs free-flow at relativistic speeds and develop a flat pressure gradient unable to transfer momentum and energy to the gas [45, 10].

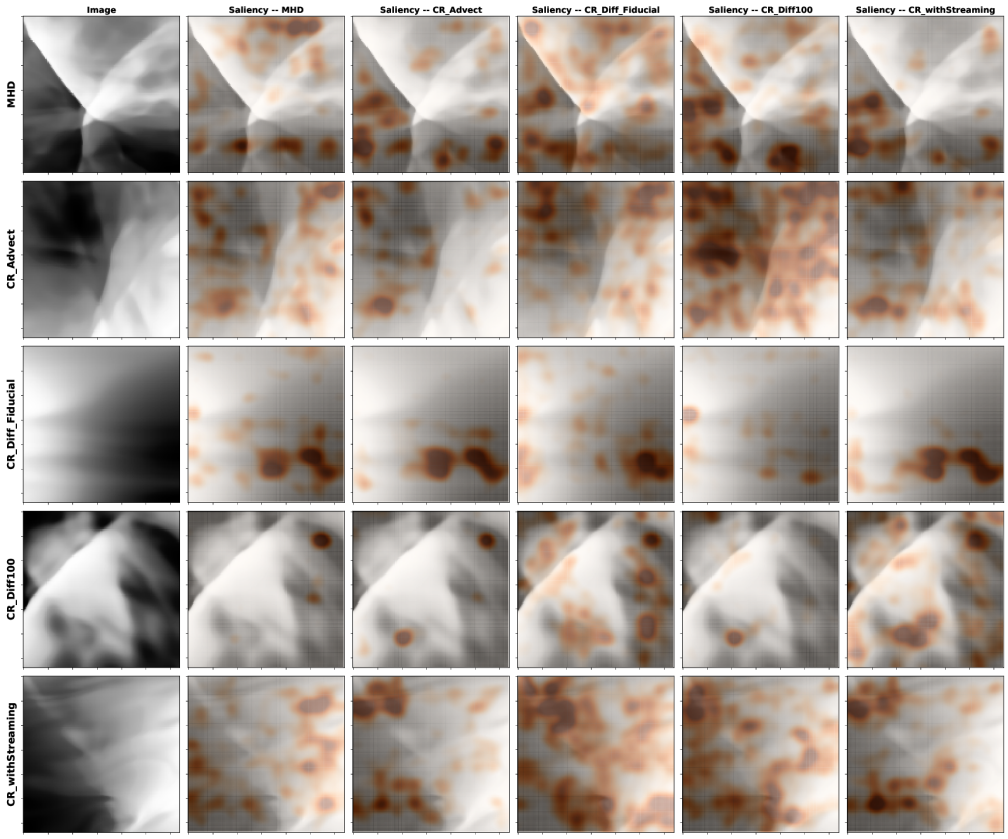

Figure 5. Saliency maps for the example `Full Power` images of Figure 1. Each row shows the same image from a given class, while each column shows activations for each of the 5 classes when the trained network is presented with that image. Activations are Gaussian smoothed instead of shown pixel-by-pixel, and they are normalized to the range [0,1], which amplifies otherwise small activations for some classes.

differences between some classes and interesting morphological overlaps between others.

3.2. Network Interpretation

We employ a combination of techniques to further interpret our results. First, we produce a set of saliency maps, which essentially show the regions of an image that produce large network activations (large gradients stored during backpropagation) leading to a final prediction. We show multiple sets of saliency maps, as different examples provide different kinds of insights. The first set of saliency maps, shown in Figure 5, is for the fiducial `Full Power` network trained on data from all 5 classes. Each row shows *one* example image from each class in grayscale (the same example images in Figure 1), and each column shows the activations from each class when presented with that image. These activations are overplotted with a white-to-red colormap. For improved readability compared to showing the pixel-by-pixel saliency, they are Gaussian filtered with kernel size = 16 and standard deviation $\sigma = 4$ (in units of number of pixels, meaning the kernel size and standard deviation are $L/4$ and $L/16$, respectively) using

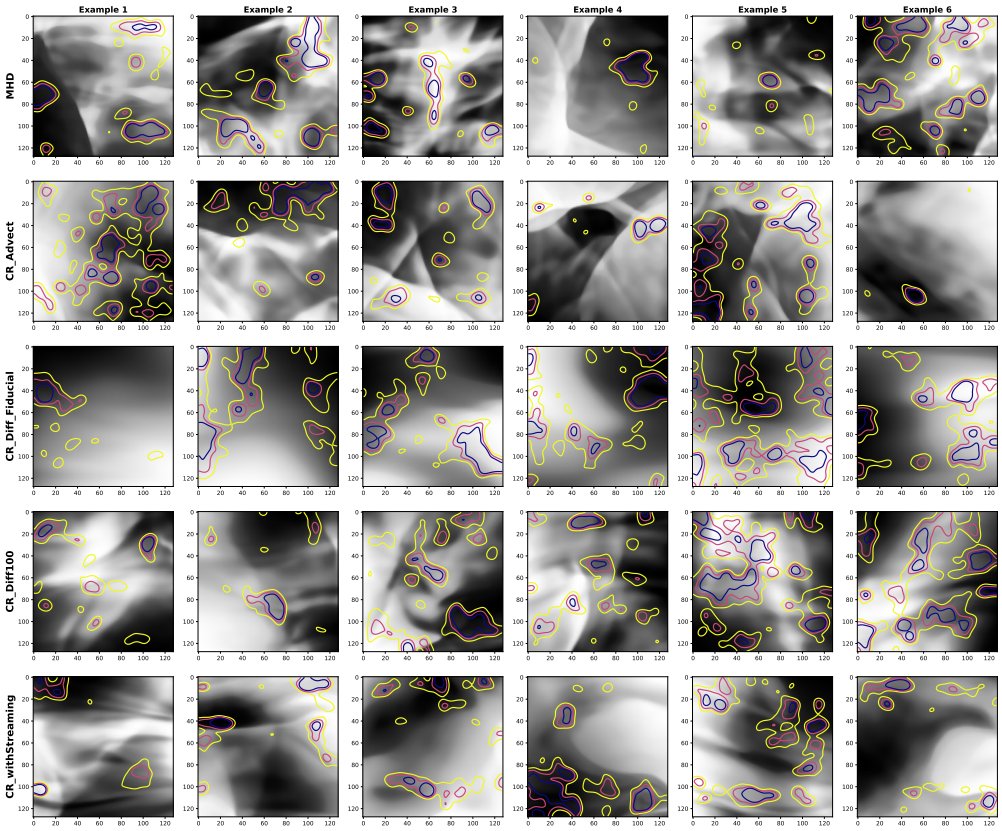

Figure 6. Saliency maps for 6 example images from each class. Yellow, red, and blue contours outline regions that triggered increasingly large activations in the neural network, i.e. the salient features of the image that influenced the network's prediction. In most cases, it is difficult to see a trend, but the `CR_Diff_Fiducial` images are generally distinguished by broad gray regions where the contrast between high and low density is smooth (large-scale) rather than sharp (small-scale).

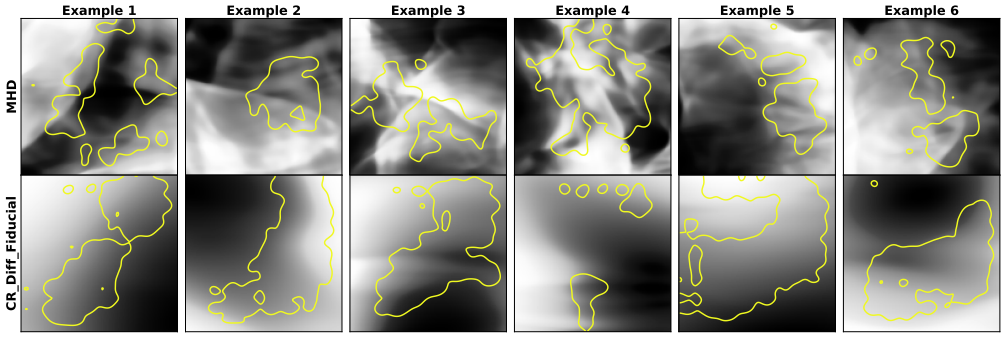

Figure 7. Image examples and overlaid saliency contours for a neural network trained on just two classes: `MHD` (top row) and `CR_Diff_Fiducial` (bottom row). Saliency contours are informative here, showing that the `MHD` images are particularly distinguished by sharp edges and regions with lots of structure, while the salient features of the `CR_Diff_Fiducial` images are broad, smooth gray regions without much small-scale structure.

the `Scipy gaussian_filter` routine. The density images themselves have not been Gaussian filtered. Note also that activations for each image are also normalized to the range [0,1]; this makes visualizing very small activations easier but misrepresents the relative magnitudes of activations for different classes.

In the second saliency figure (Figure 6), the rows of the grayscale images are for different true labels, while the columns now show 6 different examples from each labelled class. Overlaid yellow, red, and blue contours show the Gaussian-filtered, normalized saliency. Looking at enough image sets, one can convince themselves that the most salient features of the `CR_Diff_Fiducial` images are the broad, diffuse gray regions most unique to that class. Instead, sharp transitions are apparent in each of the `MHD`, `CR_Advect`, and `CR_Diff100` images, and because these features are not unique to any one class, they don't appear in our saliency maps. When we instead train a model with the same network architecture but only on the `MHD` and the `CR_Diff_Fiducial` images, saliency maps (Figure 7) make it quite obvious that what distinguishes the two classes is the sharpness of black and white transitions, i.e. the smoothness of density transitions and presence (or lack of) small-scale structure. For brevity, we omit a confusion matrix for this case, but accuracy is $> 99\%$ for both classes, showing the drastic changes imparted by diffusing CRs.

With our network trained on all 5 classes, it is hard to discern a trend that further distinguishes the `MHD`, `CR_Advect`, and `CR_Diff100`, and `CR_withStreaming` classes despite attempts to change the contour levels, etc. One possible explanation is that the higher-level, distinguishing information is imprinted as a *change in correlation over scales* rather than as a change to a local structure with well-defined boundaries, the former being very typical in physics and the latter being the typical use-case of saliency maps for e.g. object classification or detection [46, 47].

To help reveal the image features that led to predictions, we also see how the network handles a simple image manipulation: we take one test set image belonging to each class, and we Gaussian filter that image to varying extents by varying σ, the standard deviation of the Gaussian kernel. Figure 8 shows the original images in the top row, followed by the more and more Gaussian filtered images going from top to bottom. On top of each image, we denote the network's probability that the manipulated image belongs to the `CR_Diff_Fiducial` class. As σ increases, all probabilities converge to 1.0, showing that the `CR_Diff_Fiducial` images are, at least according to the network, the limit of significant "filtering" due to CR-induced damping of small-scale features.

Figure 9 shows this in a different way. The left panel shows the 1D power spectra (multiplied by k^2 to highlight differences) averaged for each class in a batch of 512 unfiltered test set images. Clearly, the `CR_Diff_Fiducial` class shows less power at intermediate scales than the `MHD` class, while the other CR classes have similar spectra. In particular, the `CR_Diff100` and `CR_withStreaming` classes have almost identical spectra, possibly leading to the confusion between those classes. The right panel shows the power spectra of `MHD` images filtered to varying extents. Unfiltered images typically have significant power at large and intermediate scales (small and intermediate

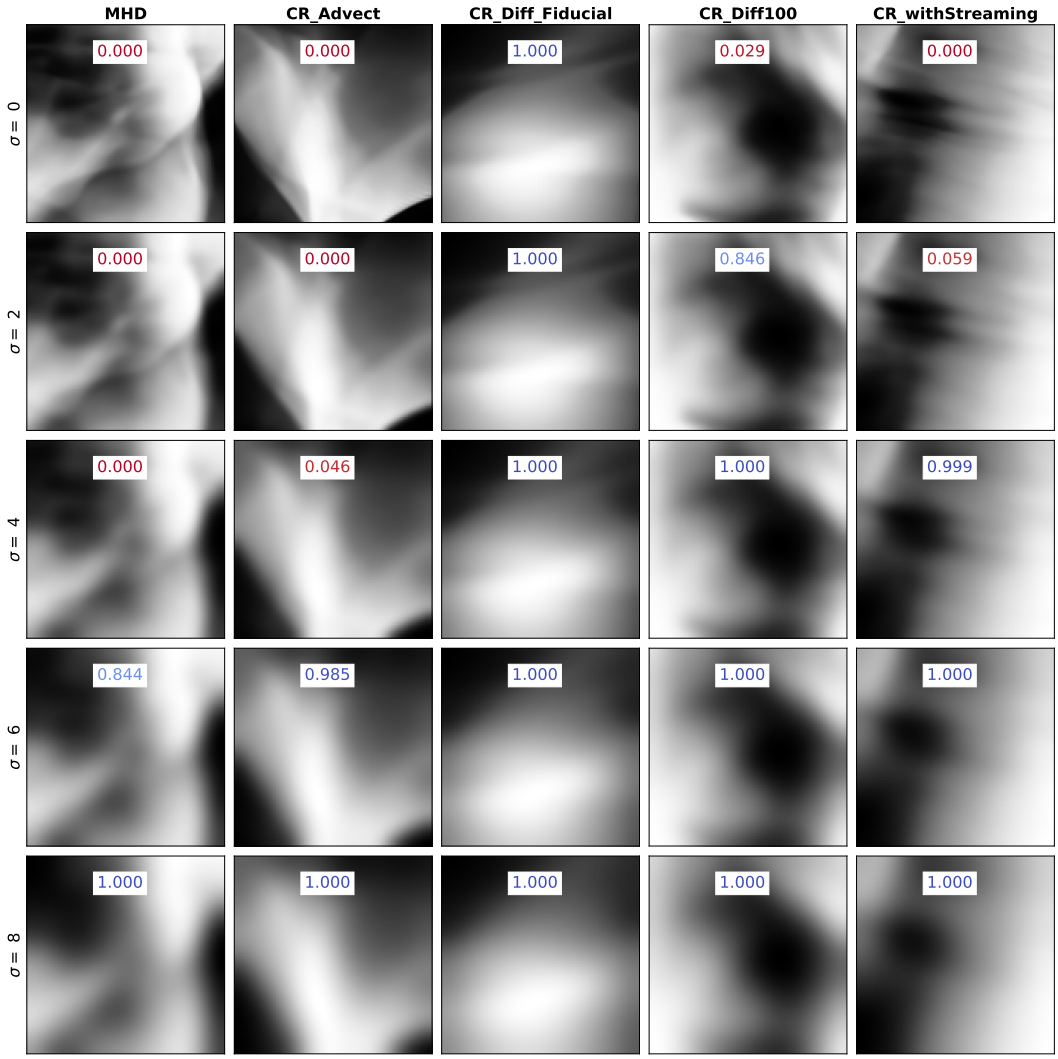

Figure 8. Experiment whereby one image from each class (different columns) is Gaussian filtered to different degrees, parametrized by different standard deviations σ of the Gaussian kernel. When run through the trained network, the output probability that the new image belongs to the `CR_Diff_Fiducial` class is denoted near the top of each image. Descending the rows (increasing σ), small-scale structure further disappears from each image, and for all classes, the images are eventually classified as `CR_Diff_Fiducial` with $> 99\%$ confidence.

wavenumbers, k) and are classified as `CR_Diff_Fiducial` with very low probability. Highly blurred images, however, lack as much power at $k < 10$ and are confidently classified as `CR_Diff_Fiducial`.

3.3. Images with Flattened Spectra

Motivated by [33], which showed that a trained CNN can distinguish sub-Alfvénic vs super-Alfvénic images even when density spectra are flattened (equaled), we flatten the power spectra of our images (the `Flattened Power` set) and train a separate

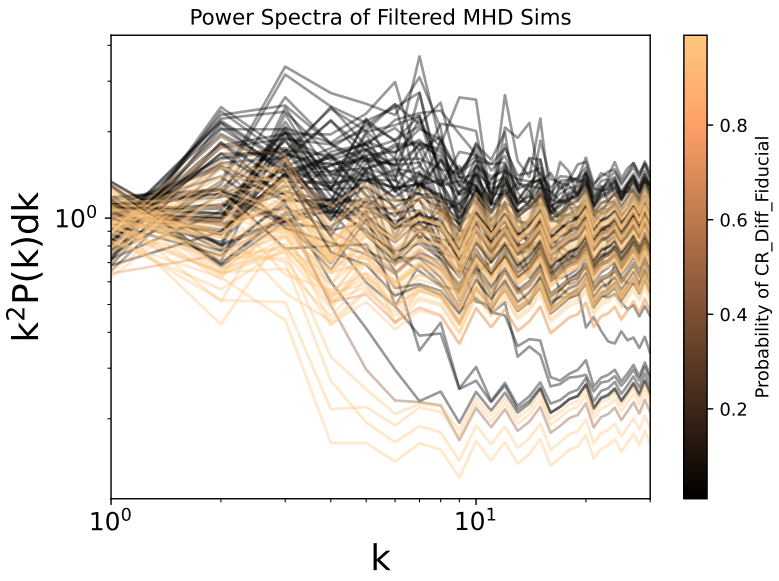

Figure 9. Power spectra of a batch of `MHD` images Gaussian filtered with varying σ. Colors denote the probability that the network classified the image as `CR_Diff_Fiducial`. Note the delineating boundary around $P(k) \sim k^{-2}$; images with decreased power at small and intermediate wavenumbers $k < 10$ are confidently classified as `CR_Diff_Fiducial`, another sign that damping is a distinguishing feature of that class.

classification network with identical architecture but different hyperparameters. This is especially interesting given what we have seen so far: that the network can learn the presence of (or lack of) spectral information, especially for the `CR_Diff_Fiducial` images, which show very little small-scale structure. However, as seen in Figure 2, spectral differences between the other classes, namely the `MHD`, `CR_Advect`, `CR_Diff100`, and `CR_withStreaming`, are quite small. This suggests the presence of other, non-spectral distinguishing features. By flattening the power spectra, we can probe this idea more explicitly and ask the network to find the salient *phase* information in the images resulting from CR interactions with gas perturbations.

The resulting confusion matrix for the test set is shown in Figure 10, and example `Flattened Power` images are shown in Figure 10. Class accuracies for the `Flattened Power` test set are comparable to those for the `Full Power` test set (see Table 2), with `CR_Diff_Fiducial` again being the most accurately predicted class, although the recall has dropped from 100% to 89.8% in exchange for a higher precision of 99.7%. F1 scores amongst the CR classes are again quite high, reaching up to 94.5% for `CR_Diff_Fiducial`. The most obvious change from the `Full Power` case is that the `MHD` and `CR_withStreaming` F1 and accuracy scores dip significantly (by more than a few percent) when their spectra are flattened. Indeed the main differences between the confusion matrices (Figures 4 and 10) are the mistaken predictions of `CR_withStreaming` when the true class is `MHD`.

To reason why these summary statistics have changed, we must consider the physical, distinguishing characteristics that might exist even in the absence of spectral information. For instance, in sub-Alfvénic turbulence, even with purely compressive forcing, spatial and spatial-temporal decompositions show that a significant portion of the energy lies in Alfvén modes [48, 49], with solenoidal motions being generated by a combination of compressive motions and magnetic forces [50]. CRs, in a transport-dependent way, have been shown to affect the ratio of solenoidal energy $E_{\rm sol}$ to compressive energy $E_{\rm comp}$ and the scale-dependent mixture of these motions (see Section 4.5 of [25]). This in-turn influences the *morphology* of density fluctuations: sharp, shock-like features indicate compressions and rarefactions, while "swirls" indicate solenoidal motions.

Bustard and Oh 2023 measure $E_{\rm sol}/(E_{\rm comp}+E_{\rm sol}) \sim 0.42$, 0.36, and 0.67 for the `MHD`, `CR_Diff_Fiducial`, and `CR_withStreaming` classes, respectively. The increased fraction of solenoidal power in the `CR_withStreaming` case should be well-imprinted in the `Flattened Power` image set, but the low precision of 58.7% for `CR_withStreaming` suggests otherwise. One possible reason is that the increase in solenoidal power is most acute at large scales $\sim L$ and almost negligible at smaller scales (see Figure 10 in [25]), meaning it might not be well-reflected in our images of size $L/2 \times L/2$. On the other hand, the `CR_Diff_Fiducial` class is well-separated from the others despite having an almost identical $E_{\rm sol}/(E_{\rm comp}+E_{\rm sol})$ to the `MHD` case. If the distinguishing image characteristics are due to solenoidal vs compressive motions, the network must be quite sensitive to them for some classes but not others. We caution, however, that even in the no-CR case, the mixture of modes in MHD turbulence is not well understood and is an active area of research. How CR transport further affects turbulent phase information and gas morphology is still a very open problem.

Alternatively, the confusion between `MHD` and `CR_withStreaming` in the `Flattened Power` set but not the `Full Power` set may simply mean the differences are largely spectrum-related rather than phase-related. As evidenced by Figure 9, there *are* some small differences between the image spectra that may have been critical in the `Full Power` case but have now been thrown out in our `Flattened Power` study.

To try to interpret our network, we again employ saliency maps on a set of example images in Figure 12. By eye, the `MHD` and `CR_Advect` image sets show long narrow features indicative of strong compression fronts, while the `CR_Diff_Fiducial` image lacks this sharp structure. However, these differences are not necessarily activated by our saliency experiment, as was the case for our `Full Power` data set. The `CR_withStreaming` activations, however, consistently line up with strong, dark line features across several example images from different classes. That these are seemingly such strong indicators of streaming CR transport is encouraging in our quest to determine the true CR transport mode in different astrophysical environments; however, these features also bear strong resemblance to the compression fronts in the `MHD` images, and significant confusion between those classes may be related to these similar features.

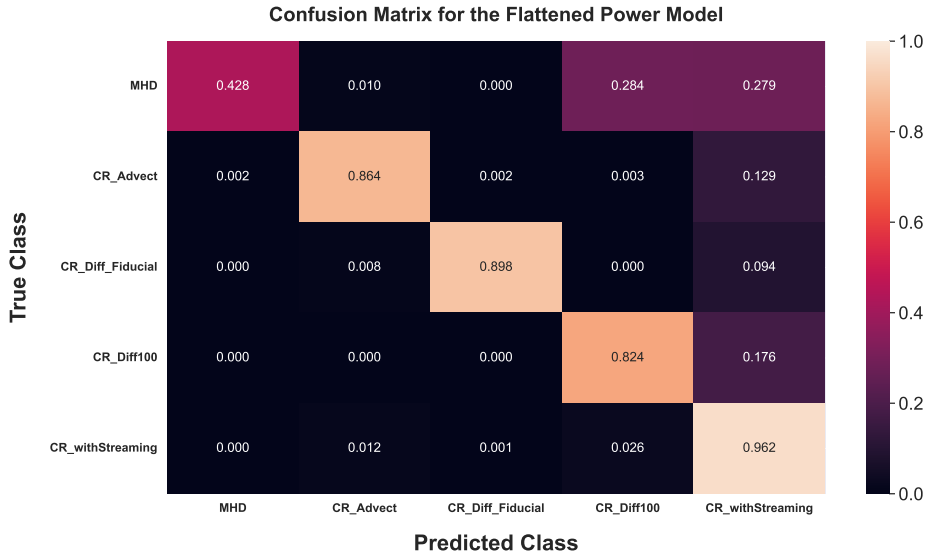

Figure 10. Confusion matrix for the `Flattened Power` set of test images. Accuracies range from $\sim 85 - 98\%$; As all images have flattened power spectra now, clearly, CRs impart distinguishing *phase* changes to the turbulent flow that the network appears to learn. The MHD case again is the least reliable, with a low recall of 42.8% and frequent confusion with the `CR_Diff100` and `CR_withStreaming` classes.

We experimented with a few other interpretability tools, most notably occlusion experiments where one asks the network to predict an image with obscured regions (ideally regions where salient features are present), but no experiments to-date have gleamed much new information. For brevity, we stop our exploratory analysis here, but follow-up studies can probe the origin of distinguishing, non-spectral features, which are possibly related to the CR transport-dependent mixture of solenoidal and compressive motions found in Bustard and Oh 2023. Related work using wavelet scattering transforms, which construct similar representations as CNNs (e.g. [51, 52, 53]), has demonstrated the importance of encoding phase information and scale separation. Additional image manipulations and explicitly adding image spectra as an extra input to the `Flattened Power` model would be useful next steps to reveal and isolate additional distinguishing features and gain deeper insights on CR-induced differences.

4. Limitations and Future Work

4.1. Simulation Limitations

While our results are promising, this method is not without its limitations. The biggest issue we hope to tackle in future work is domain adaptation: can this network, trained on simulation data, generate accurate predictions when deployed on real observations? For this supervised learning algorithm to back out CR transport from observations, we require the "ground truth" given to us by simulations, but the simulations have a number of restrictions that could limit our machine learning model's ability to generalize

to observations:

(i) **Physics choices:** For instance, these simulations use an isothermal equation of state instead of an adiabatic equation of state with realistic radiative cooling, conduction, self-gravity, etc. Additionally, ensuring the correct ionization state of the gas is crucial to making a robust connection to observations, but this is very environment-specific and dependent on the local distribution of nearby massive stars, etc.

(ii) **Simulation parameter coverage:** For instance, the results shown here are trained on a simulation suite with a number of parameters fixed – the stirring rate, the initial plasma beta, etc. One could increase the span of the training data across different parameters, but the data volume would quickly become very large.

(iii) **Simulation convergence:** Bustard and Oh 2023 conducted a limited convergence study, showing that the main transport-dependent trends (CR-induced damping of small-scale fluctuations, etc.) are robust to changes in resolution; however, increasing resolution will always lead to more small-scale structure because the turbulent inertial range, which artificially dissipates in simulations on length scales of ≈ 30 cell widths [54], extends to smaller scales. Higher resolution simulations, being more computationally intensive, are not possible at this point, but in the future, one might create a more robust network by training it on simulations with varying resolution.

One could also map simulations to observations more closely by folding in additional telescope effects during preprocessing, but this depends on the case-by-case telescope instrumentation and is beyond the scope of this paper.

4.2. Sensitivity to Projection Depth

Alleviating the issues above will require additional simulations. With our existing simulation suite, though, we can quickly explore another limitation: our use of single-cell-thick slice plots as training data, rather than projections that integrate further along our line-of-sight. To test this, we create two additional image sets by averaging over d cells perpendicular to the image plane, rather than creating images from single-cell slices. The resulting test sets have roughly $3,800$ and 760 images per class for $d = 8$ and 32, respectively.

Figure 11 shows confusion matrices when our model (pre-trained on slices) is asked to predict images with $d = 8$ and 32. In both cases, the model can very accurately classify the `CR_Diff_Fiducial` images, but as the depth increases, overall accuracy decreases significantly. This is most true for the `MHD` images, which show the crux of the issue: averaging / projecting over multiple layers smooths out the small-scale structure that distinguishes the other classes, particularly the `MHD` class, from the others.

If there is a significant gap between the training projection depth and the test projection depth, then accuracy can degrade significantly. This is not entirely

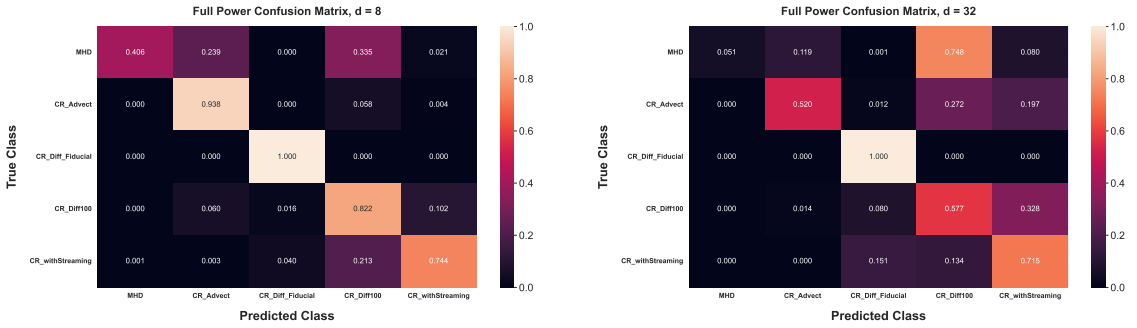

Figure 11. Confusion matrix for our pre-trained `Full Power` model applied to images created by averaging over $d = 8$ cells (left) and $d = 32$ cells (right) along the $\hat{x}$-axis. Note how accuracies decrease significantly, particularly for classes that originally were characterized by sharp density gradients and small-scale structure; these salient features are smoothed out when averaging over more and more layers, leading to numerous misclassifications.

unexpected, since this "model misspecification" or "domain shift," i.e. applying a model trained on one dataset to another, is an active and unsolved area of research (although there have been recent promising results in domain adaptation for astronomical machine learning; e.g. see [55]). The issue of projection depth also hinders direct application of our method to observational data sets, as distances to interstellar clouds in the Milky Way still have significant uncertainties [56].

Overall, given the critical differences between current simulations and real observations, challenges associated with observational uncertainties, and the problem of domain shift in supervised machine learning, we caution against directly applying our method to real data. However, our publicly available code can serve as a useful framework for data preprocessing and model training, while our publicly available data volumes, which are scale-free and can therefore be scaled up or down to different astrophysical regimes depending on the turbulent driving scale L_0, can serve as a useful comparison to more detailed simulations in the future.

5. Conclusions

Deducing CR transport physics from observations is a massive undertaking, involving phenomenological models fit to direct and indirect CR observables [1], galaxy and zoom-in cosmological simulations compared to radio synchrotron and gamma-ray emission [57, 58], and focused probes of CR penetration in cold molecular clouds [59, 60, 61, 16] and CR transport along radio-emitting filaments [62]. All of these research avenues rely on multi-wavelength data, e.g. high-energy emission from radio emitting CR electrons or gamma-ray emission arising from hadronic interactions of thermal gas with CR protons. An alternative, and to our knowledge unexplored, avenue is to harness recent advances in deep learning and a growing amount of simulation data to train a network to recognize CR transport physics from solely density images.

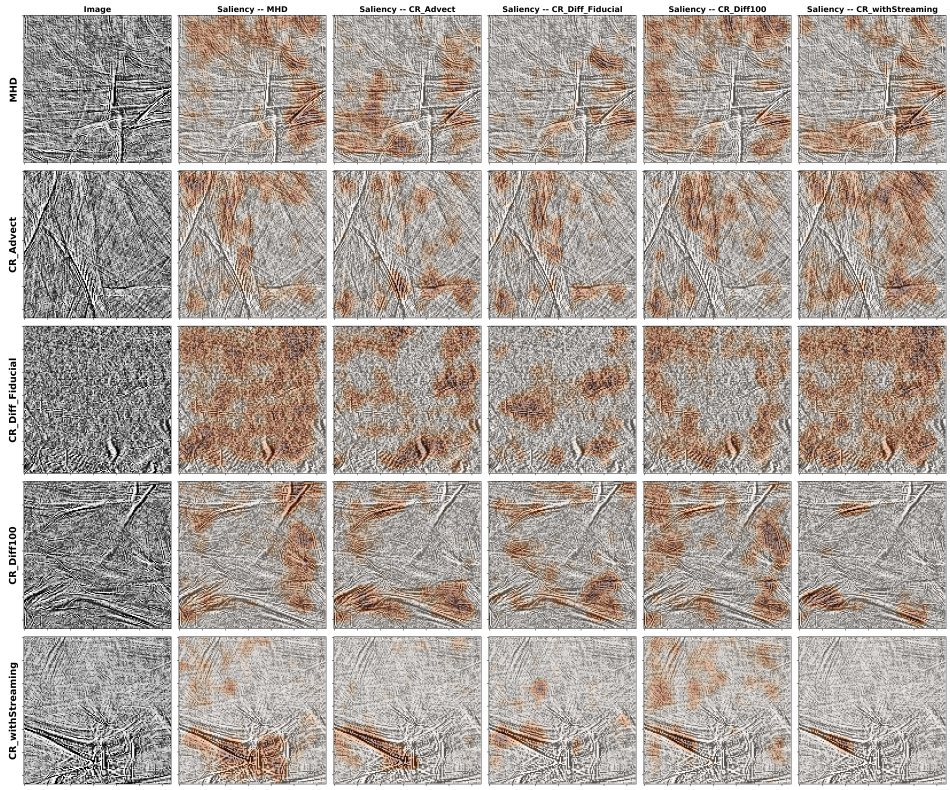

Figure 12. Same as Figure 5 but for example images from the `Flattened Power` test set. Many of the same regions are activated for each class; however, when looking at the activations for the true class, it's apparent in some cases that certain, isolated features drive the prediction. See e.g. `CR_withStreaming`, in which case the true prediction map ("saliency – `CR_withStreaming`") shows a high activation in only one small area, whereas activations for other classes are more spread out.

In this paper, we trained and fine-tuned multi-layer convolutional neural networks (CNNs) on a suite of turbulent box simulations [25] with varying CR transport prescriptions from pure CR advection to CR diffusion to CR streaming. We also use interpretability tools like saliency maps and image manipulation to interpret these results and to help build physical intuition for CR impacts on turbulence.

The main findings of this work are:

- Our trained CNN can classify images originating from simulations with 5 different CR transport prescriptions with high class accuracies ranging from 92.0% to 99.2% and F1 scores ranging from 80.8% to 98.0%. The average recall is brought down most significantly by the MHD-only (no CR) simulations, whose resulting density images closely resemble those with either very little or very fast CR transport.
- Images derived from simulations with intermediate diffusivity, i.e. `CR_Diff_Fiducial`, are most accurately classified (99.2%). Saliency maps (Figures 6 and 7) identify smooth, rather than sharp, density contrasts as distinguishing features of these images, owing to strong CR-induced damping of small-scale turbulent fluctuations [25] that effectively Gaussian filters or "blurs" the image (Figure 8).

- Streaming and diffusion lead to distinctly different images and are only rarely confused by our trained network (Figure 4); however, there is some confusion between streaming transport and fast diffusive transport.
- Images with flattened power spectra are also classified with high accuracies (85.7-97.9%), suggesting that CRs change both spectral *and* phase information, as would be the case if CRs affect the balance between compressive and solenoidal motions as found in [25]. In particular, saliency maps (Figure 12) reveal that the network consistently associates streaming transport with strong, dark lines in spectrally flattened gas density images.

Acknowledgments

The authors gratefully acknowledge Peng Oh, Josh Peek, and Blakesley Burkhart for stimulating discussions, as well as the organizers and participants of the KITP program "Building a Physical Understanding of Galaxy Evolution with Data-driven Astronomy" where this work originated. This research was supported in part by the NSF PHY-2309135 grant to the KITP. CB was supported by the National Science Foundation under Grant No. NSF PHY-1748958 and by the Gordon and Betty Moore Foundation through Grant No. GBMF7392. Turbulence simulations were performed on the Stampede2 supercomputer under allocation TG-PHY210004 provided by the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1548562 [63].

References

[1] Hanasz M, Strong A W and Girichidis P 2021 *Living Reviews in Computational Astrophysics* **7** 2
[2] Becker Tjus J and Merten L 2020 Phys. Rep. **872** 1–98
[3] Zweibel E G 2017 *Physics of Plasmas* **24** 055402
[4] Ruszkowski M and Pfrommer C 2023 *arXiv e-prints* arXiv:2306.03141
[5] Yan H and Lazarian A 2004 ApJ **614** 757–769
[6] Wentzel D G 1968 ApJ **152** 987
[7] Kulsrud R and Pearce W P 1969 ApJ **156** 445
[8] Boulares A and Cox D P 1990 ApJ **365** 544
[9] Simpson C M, Pakmor R, Marinacci F, Pfrommer C, Springel V, Glover S C O, Clark P C and Smith R J 2016 ApJ **827** L29
[10] Wiener J, Oh S P and Zweibel E G 2017 MNRAS **467** 646–660
[11] Pfrommer C, Pakmor R, Schaal K, Simpson C M and Springel V 2017 MNRAS **465** 4500–4529
[12] Ruszkowski M, Yang H Y K and Zweibel E 2017 ApJ **834** 208
[13] Buck T, Pfrommer C, Pakmor R, Grand R J J and Springel V 2020 MNRAS **497** 1712–1737
[14] Hopkins P F, Chan T K, Garrison-Kimmel S, Ji S, Su K Y, Hummels C B, Kereš D, Quataert E and Faucher-Giguère C A 2020 MNRAS **492** 3465–3498
[15] Ji S, Chan T K, Hummels C B, Hopkins P F, Stern J, Kereš D, Quataert E, Faucher-Giguère C A and Murray N 2020 MNRAS **496** 4221–4238
[16] Bustard C and Zweibel E G 2021 ApJ **913** 106
[17] Huang X, Jiang Y f and Davis S W 2022 ApJ **931** 140
[18] Faucher-Giguère C A and Oh S P 2023 ARA&A **61** 131–195

[19] Bustard C, Zweibel E G, D'Onghia E, Gallagher J S I and Farber R 2020 ApJ **893** 29
[20] Ptuskin V S 1981 Ap&SS **76** 265–278
[21] Bustard C and Oh S P 2022 ApJ **941** 65
[22] Begelman M C and Zweibel E G 1994 ApJ **431** 689
[23] Tsung T H N, Oh S P and Jiang Y F 2022 MNRAS **513** 4464–4493
[24] Quataert E, Jiang Y F and Thompson T A 2022 MNRAS **510** 920–945
[25] Bustard C and Oh S P 2023 ApJ **955** 64
[26] Ptuskin V S 1988 *Soviet Astronomy Letters* **14** 255
[27] Brunetti G and Lazarian A 2011 MNRAS **410** 127–142
[28] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Kopf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J and Chintala S 2019 Pytorch: An imperative style, high-performance deep learning library *Advances in Neural Information Processing Systems 32* (Curran Associates, Inc.) pp 8024–8035 URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf
[29] Strong A W, Moskalenko I V and Ptuskin V S 2007 *Annual Review of Nuclear and Particle Science* **57** 285–327
[30] Stone J M, Tomida K, White C J and Felker K G 2020 ApJS **249** 4
[31] Jiang Y F and Oh S P 2018 ApJ **854** 5
[32] Uhlenbeck G E and Ornstein L S 1930 *Physical Review* **36** 823–841
[33] Peek J E G and Burkhart B 2019 ApJ **882** L12
[34] Goldreich P and Sridhar S 1995 ApJ **438** 763
[35] van der Walt S, Schönberger J L, Nunez-Iglesias J, Boulogne F, Warner J D, Yager N, Gouillart E, Yu T and the scikit-image contributors 2014 *PeerJ* **2** e453 ISSN 2167-8359
[36] Huertas-Company M and Lanusse F 2023 PASA **40** e001
[37] Loshchilov I and Hutter F 2017 *arXiv e-prints* arXiv:1711.05101
[38] Kingma D P and Ba J 2014 *arXiv e-prints* arXiv:1412.6980
[39] Ramachandran P, Zoph B and Le Q V 2017 *arXiv e-prints* arXiv:1710.05941
[40] Ramachandran P, Zoph B and Le Q V 2017 Searching for activation functions (*Preprint* 1710.05941)
[41] Misra D 2020 Mish: A self regularized non-monotonic activation function (*Preprint* 1908.08681)
[42] Guo Z, Wu J F and Sharon C E 2022 *arXiv e-prints* arXiv:2212.07881
[43] Wu J F, Peek J E G, Tollerud E J, Mao Y Y, Nadler E O, Geha M, Wechsler R H, Kallivayalil N and Weiner B J 2022 ApJ **927** 121
[44] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M and Duchesnay E 2011 *Journal of Machine Learning Research* **12** 2825–2830
[45] Skilling J 1971 ApJ **170** 265
[46] Selvaraju R R, Cogswell M, Das A, Vedantam R, Parikh D and Batra D 2016 *arXiv e-prints* arXiv:1610.02391
[47] Meng F, Huang K, Li H and Wu Q 2019 *arXiv e-prints* arXiv:1901.07683
[48] Makwana K D and Yan H 2020 *Physical Review X* **10** 031021
[49] Gan Z, Li H, Fu X and Du S 2022 ApJ **926** 222
[50] Lim J, Cho J and Yoon H 2020 ApJ **893** 75
[51] Bruna J and Mallat S 2013 *IEEE transactions on pattern analysis and machine intelligence* **35** 1872–1886
[52] Cheng S, Morel R, Allys E, Ménard B and Mallat S 2023 *arXiv e-prints* arXiv:2306.17210
[53] Velicheti P D, Wu J F and Petric A 2023 PASP **135** 084502
[54] Federrath C, Roman-Duval J, Klessen R S, Schmidt W and Mac Low M M 2010 A&A **512** A81
[55] Ćiprijanović A, Lewis A, Pedro K, Madireddy S, Nord B, Perdue G N and Wild S M 2023 *Machine Learning: Science and Technology* **4** 025013

[56] Green G M, Schlafly E, Zucker C, Speagle J S and Finkbeiner D 2019 ApJ **887** 93
[57] Chan T K, Kereš D, Hopkins P F, Quataert E, Su K Y, Hayward C C and Faucher-Giguère C A 2019 MNRAS **488** 3716–3744
[58] Hopkins P F, Squire J, Butsky I S and Ji S 2022 MNRAS **517** 5413–5448
[59] Everett J E and Zweibel E G 2011 ApJ **739** 60
[60] Morlino G and Gabici S 2015 MNRAS **451** L100–L104
[61] Dogiel V A, Chernyshov D O, Ivlev A V, Malyshev D, Strong A W and Cheng K S 2018 ApJ **868** 114
[62] Thomas T, Pfrommer C and Enßlin T 2020 ApJ **890** L18
[63] Towns J, Cockerill T, Dahan M, Foster I, Gaither K, Grimshaw A, Hazlewood V, Lathrop S, Lifka D, Peterson G D, Roskies R, Scott J R and Wilkins-Diehr N 2014 *Computing in Science & Engineering* **16** 62–74 ISSN 1521-9615