CosmoFlow: Scale-Aware Representation Learning for Cosmology with Flow Matching

Sidharth Kannan [1 2] **Tian Qiu** [3] **Carolina Cuesta-Lazaro** [4 5 6] **Haewon Jeong** [3]

Abstract

Generative machine learning models have been demonstrated to be able to learn low dimensional representations of data that preserve information required for downstream tasks. In this work, we demonstrate that *flow matching*-based generative models can learn compact, semantically rich latent representations of field level cold dark matter (CDM) simulation data without supervision. Our model, **CosmoFlow**, learns representations 32x smaller than the raw field data, usable for field level reconstruction, synthetic data generation, and parameter inference. Our model also learns *interpretable* representations, in which different latent channels correspond to features at different cosmological scales.

1. Introduction

The large-scale structure of the Universe provides one of the most stringent tests of gravity on cosmological scales. Over the past decades, the ΛCDM cosmological model has emerged as the standard framework for understanding our cosmos, where Λ represents the cosmological constant (associated with dark energy) and CDM denotes cold dark matter—which together comprise approximately 95% of the Universe's energy budget. Theoretical predictions of ΛCDM can now be implemented with remarkable precision in numerical simulations, which capture the formation of the cosmic web: an intricate network where galaxies reside in dense clusters, connected by filamentary structures and separated by vast cosmic voids.

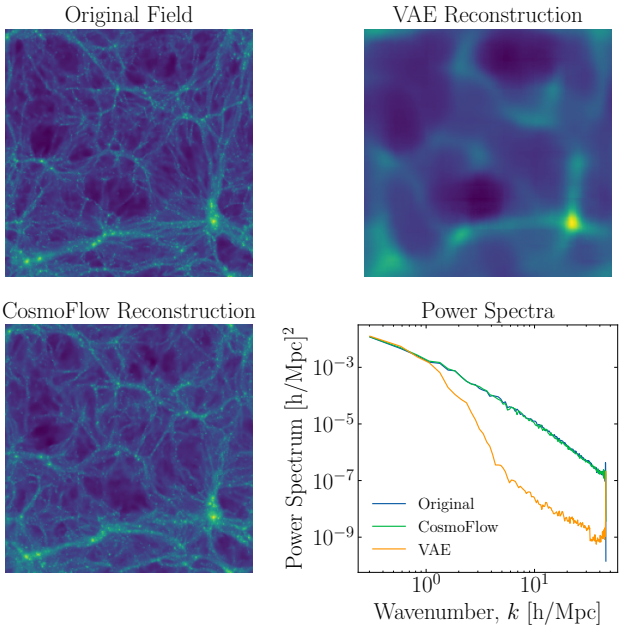

Figure 1. We compare reconstruction quality of our model, CosmoFlow, to the reconstructions produced by a VAE with the same size latent code. While standard VAEs produce blurry reconstructions, our model is able to capture fine detail. We note that the model still deviates from the ground truth at high frequencies.

This success, however, presents cosmology with a new challenge. High-resolution simulations like AbacusSummit generate datasets exceeding 2000 TB, severely constraining our ability to scale training datasets for machine learning applications. Moreover, extracting meaningful insights from these high-dimensional datasets requires models that can effectively navigate the curse of dimensionality.

Representation learning offers a promising solution by mapping high-dimensional simulation data into low-dimensional representations suitable for downstream tasks such as cosmological parameter inference, anomaly detection, i.e., looking for deviations to ΛCDM, and data compression. However, the application of representation learning to cosmology faces a fundamental tension between two competing objectives: faithful data reconstruction and semantic information extraction.

[1]College of Creative Studies, University of California, Santa Barbara, Santa Barbara, USA [2]Department of Physics, University of California, Santa Barbara, Santa Barbara, USA [3]Department of Electrical and Computer Engineering, University of California, Santa Barbara, Santa Barbara, USA [4]The NSF AI Institute for Artificial Intelligence and Fundamental Interactions, Cambridge, MA 02139, USA [5]Department of Physics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA [6]Center for Astrophysics —Harvard & Smithsonian, 60 Garden St, Cambridge, MA 02138, USA. Correspondence to: Sidharth Kannan <skannan@ucsb.edu>.

Traditional compression prioritizes minimizing information loss for perfect reconstruction, but many cosmological analyses do not require pixel-level fidelity. Instead, the goal is to capture scientifically relevant information—analogous to how the power spectrum discards spatial phase information while preserving the statistical properties crucial for parameter inference.

Existing approaches to cosmological representation learning fall into two broad categories. Contrastive approaches learn representations by distinguishing between positive and negative example pairs, requiring explicit definitions of which examples should be proximate or distant in latent space. For instance, [Akhmetzhanova et al. (2023)] defines positive pairs as simulations sharing identical cosmological parameters but differing in initial conditions, while negative pairs correspond to simulations with different cosmological parameters.

Generative approaches, conversely, learn representations by reconstructing the original data distribution from the latent space. While Variational Autoencoders (VAEs) [(Kingma & Welling, 2022)] have been widely adopted for this purpose and achieve strong downstream task performance, they typically produce blurry reconstructions that lose critical high-frequency details. Unlike natural images where high frequency details can be perceptually insignificant, a significant amount of cosmological information is present in the small scale structure.

We adopt a generative framework for the following reasons. First, generative methods eliminate the need for assumptions about what constitutes positive and negative pairs, whereas in constrastive approaches the choice of pairing strategy can substantially impact learned representations. Second, generative models serve multiple purposes: they enable data compression through dimensionality reduction, facilitate fast generation of new, synthetic data, and produce semantically rich representations suitable for downstream cosmological analyses.

Recently, the flow matching paradigm [Lipman et al. (2023; 2024); Albergo et al. (2023)] has been demonstrated to achieve state-of-the-art performance for generation across image, audio, and other domains. A flow matching model learns to map a noise sample to a data sample, via a time-dependent vector field. In this paper, we apply flow matching to the representation learning problem for cosmology, and demonstrate that it enables the learning of useful and interpretable latent representations.

We summarize our contributions as follows:

- We present CosmoFlow, the first cosmological representation learning model usable for both high quality reconstruction and downstream tasks. We show that our model is able to compress 256×256 pixel field data down to an 8 element vector that can be used to estimate the cosmological parameters with equivalent accuracy as estimation on the raw field data.
- We show that our model can be used to generate reconstructions of field data from a latent 32x smaller than the original images, and to generate new, synthetic data for parameter values not in the dataset.
- We demonstrate that the inductive biases of flow matching can be used to build a latent space where different parts of the representation correspond to features at different cosmological scales.

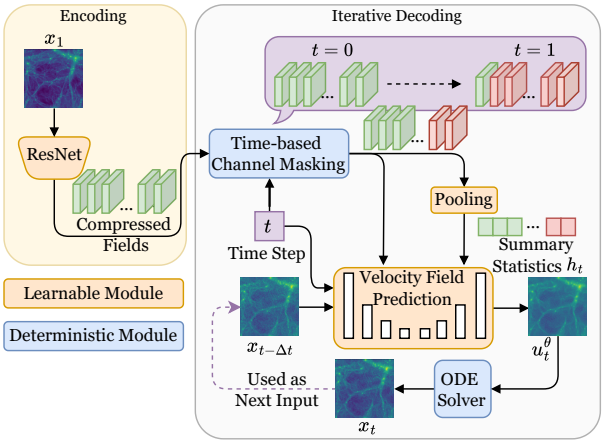

Figure 2. **An overview of CosmoFlow**. A ResNet encodes the input image to compressed fields. During each time step of iterative decoding, the compressed field is masked and passed through a global pooling layer to generate a compact summary statistics vector. Both the masked compressed field and the summary statistics are used as conditions for the UNet-based velocity field prediction. See more details in Sec. 3.

2. Background and Related Work

Flow Matching Flow matching [Lipman et al. (2023)] is a framework for generative modelling, closely related to continuous normalizing flows [Chen et al. (2018)] and the diffusion family of models [Song et al. (2021; 2020)]. In flow matching, data is mapped from a prior distribution to the data distribution via a *probability flow*, defined via a time-dependent vector field, called the *velocity field*. Samples are then generated by solving the probability flow ordinary differential equation (ODE), with an initial condition X_0, randomly sampled from the prior:

$$\frac{d}{dt} X_t = u_t^\theta(X_t). \tag{1}$$

While in general, flow matching allows the prior distribution to take any form, in this work, we use the standard Gaussian, $X_0 \sim \mathcal{N}(0, \mathbf{I})$. During training, the model attempts to learn a velocity field that maps a sample from the prior to the

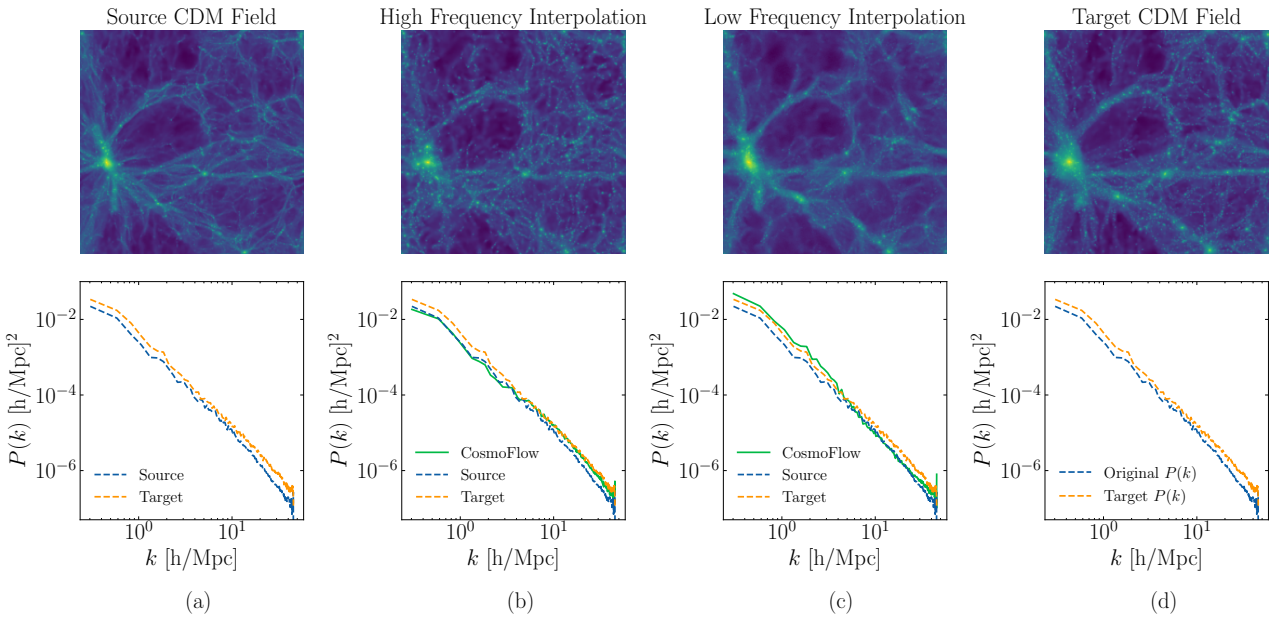

Figure 3. (a) The initial CDM field (b) We interpolate the channels corresponding to high frequencies. The generated image contains many more granular features. Only the high frequency components in the power spectrum rise. (c) We interpolate the channels corresponding to low frequencies. The image smooths, and only the low frequency components in the power spectrum rise.(d) The target CDM field data.

training sample. This gives rise to the conditional flow matching loss,

$$\mathcal{L}_{\text{CFM}} = \mathbb{E}_{t,q(X_1),p_t(X|X_1)}||v_t(x) - u_t(X|X_1)||_2^2. \quad (2)$$

In our case, we choose the mapping to be a straight line path. Then, the flow matching loss can be written as a regression loss, between the model output and the difference between the training sample and initial condition:

$$\mathcal{L} = \frac{1}{N}\sum_{n}^{N} ||u_t^\theta(X_t, t) - (X_1 - X_0)||^2. \quad (3)$$

The CAMELS Dataset The CAMELS Multifield Dataset [Villaescusa-Navarro et al. (2021)] is a dataset comprised of thousands of hydrodynamic simulations across a wide range of cosmological and astrophysical parameters. The primary goal of CAMELS is to elucidate the connection between the various cosmological and astrophysical parameters, and observable features of the universe by enabling the training of machine learning models.

CAMELS contains multiple different simulation suites, each with a different implementation of small scale physics. Each simulation suite captures a range of cosmological fields, including the dark matter distribution, gas distribution, temperature, etc. We train our model using cold dark matter maps from the Astrid hydrodynamic simulation suite [Ni et al. (2023)].

Representation Learning in Cosmology Recent work in cosmological representation learning has explored both generative and contrastive paradigms. Andrianomena & Hassan (2023) developed VAEs for cosmological fields, showing strong performance in parameter inference but suffering from the characteristic VAE limitation of blurry reconstructions. Akhmetzhanova et al. (2023) focused on contrastive approaches, defining positive pairs as simulations with identical cosmologies but different initial conditions, and negative pairs as simulations with different cosmologies.

3. Method

Model Architecture CosmoFlow is composed of two parts: a) a ResNet [He et al. (2016)] based encoder, which produces a lower dimensional representation of the input, used to condition the decoder, and b) a UNet [Ronneberger et al. (2015)] based decoder, which attempts to reconstruct the input image through velocity field estimation. The details of the architecture are shown in Fig. 2.

Learning Spatially Meaningful Latents with Progressive Masking Our goal is to design the latent space such that different channels in the compressed field correspond to different scales in the reconstructed image. We do this by employing a version of the framework from Yue et al. (2024). A channel-wise mask is applied to the compressed field such that at $t = 0$ (start of generation), all channels are unmasked. As t increases, latent channels are progressively masked out, until at $t = 1$, only one remains. This approach is inspired by the inference process in the flow matching family of models (see Appendix B), in which a noise sample is iteratively denoised. Low frequency, large scale structure is reconstructed first, while the smaller scale features are

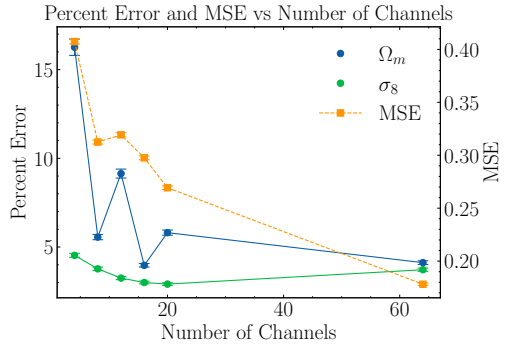

Figure 4. We show the effect of varying the number of channels in the compressed field on parameter inference and reconstruction quality. Error bars show the standard error across the validation set.

refined closer to $t = 1$. Thus, the latent channel that remains at $t = 1$ encodes the latent information corresponding to the highest frequency.

4. Results

Reconstruction We demonstrate that our model is able to produce significantly higher fidelity reconstructions than other standard generative models. In particular, we compare against a VAE with the same size latent. We show that CosmoFlow achieves significantly more realistic reconstructions, which is also reflected in the power spectra plot—VAE loses high-frequency information while CosmoFlow preserves all frequencies (see Fig. 1). We note, however, that our models reconstructions are still lossy; while they are realistic, there are still deviations at high frequency, which can be seen by comparing filaments between the original and reconstruction.

Parameter Inference One of the primary applications of representation learning models to cosmology is to learn low dimensional representations which still allow estimation of the posterior distribution of cosmological parameters. We demonstrate that the *summary statistics*, produced by the encoder can be used for parameter inference with similar accuracy to inference on the raw field data. In this work, we focus on predicting the posterior mean of the cosmological parameters instead of the full distribution.

As a baseline, we train a ResNet-18 for parameter inference using the raw field data. This achieves 4.96% and 2.94% mean relative errors for Ω_m and σ_8 respectively. We did not spend much time on hyperparameter optimization for this network, so it is possible that marginally better results could be achieved. We achieve 5.24% and 4.03% using the 8 channel version of our model. We highlight that the information bottleneck here is just 8 floating point numbers, as compared to the 65,536 pixels in the original field data. The full output of the encoder (*compressed field* in Fig. 2) is not used for parameter inference, only the 8 element summary statistics. The 16-channel model's latents achieve 3.72% and 3.00% mean relative errors, but exhibits worse frequency disentanglement. We plot the results of parameter inference as we vary the number of latent channels in Fig. 4.

Anomaly Detection Another task of interest is *anomaly detection*. In particular, we attempt to distinguish between cold dark matter and warm dark matter (WDM) maps using summary statistics. We observe that a) the model takes input WDM images, and then converts them to be in-distribution, CDM maps, as illustrated in Fig. 5 and b) the latent representations of WDM maps are not easily separable from the latent representations of CDM maps.

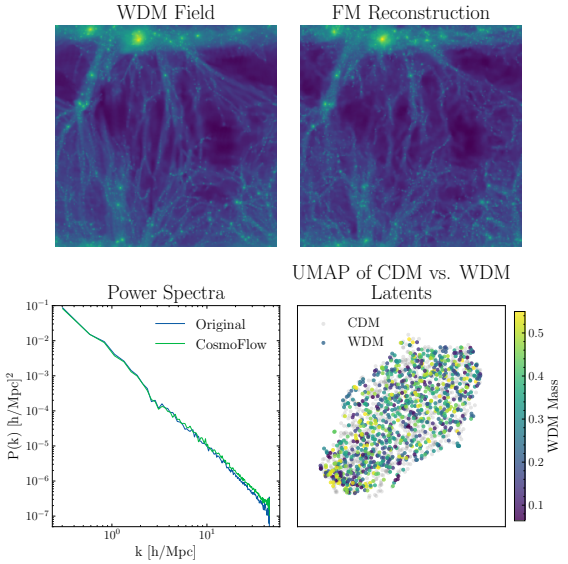

Figure 5. Warm dark matter maps are converted to "nearest" cold dark matter maps. In particular, we observe CosmoFlow artificially adds in fine structure, and the power spectrum over-represents high frequencies. The latent representations of CDM and WDM are not separable by UMAP.

Frequency-Based Interpolation/Latent Space Disentanglement Our model is designed to provide a disentangled representation, where different channels in the compressed field correspond to different spatial scales in the reconstruction. This gives us the ability to modulate the large scale structures *independently* of the fine details, simply by interpolating the latent channels corresponding to those frequencies (See Fig. 3). As we increase the number of latent channels, this separability degrades; this is likely due to the latent space having "too much" capacity, leading to latent channels encoding somewhat redundant information.

5. Discussion and Future Work

In this work, we demonstrate the utility of flow matching for learning representations of CDM simulation data. However, our current representations do not effectively capture fea-

tures that distinguish CDM from WDM fields, and improving this remains a key goal, particularly to enable anomaly detection. We also plan to evaluate our representations in transfer learning settings, investigating whether they can be fine-tuned for new datasets with limited samples. To support compression and reconstruction, we aim to incorporate neural compression modules [Ballé et al. (2018); Yang & Mandt (2023)], which may help reduce data size further. Finally, we see potential in extending the flow matching approach to other data modalities, such as directly operating on raw point cloud data instead of field-level maps.

6. Software

The code used to produce these results is accessible at https://github.com/sidk2/cosmo-compression.

References

Akhmetzhanova, A., Mishra-Sharma, S., and Dvorkin, C. Data compression and inference in cosmology with self-supervised machine learning. *Monthly Notices of the Royal Astronomical Society*, 527(3):7459–7481, November 2023. ISSN 1365-2966. doi: 10.1093/mnras/stad3646. URL http://dx.doi.org/10.1093/mnras/stad3646.

Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

Albergo, M. S., Boffi, N. M., and Vanden-Eijnden, E. Stochastic interpolants: A unifying framework for flows and diffusions, 2023. URL https://arxiv.org/abs/2303.08797.

Andrianomena, S. and Hassan, S. Latent space representations of cosmological fields, 2023. URL https://arxiv.org/abs/2311.00799.

Ballé, J., Laparra, V., and Simoncelli, E. P. End-to-end optimized image compression. *arXiv preprint arXiv:1611.01704*, 2016.

Ballé, J., Minnen, D., Singh, S., Hwang, S. J., and Johnston, N. Variational image compression with a scale hyperprior. *arXiv preprint arXiv:1802.01436*, 2018.

Chen, R. T., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hudson, D. A., Zoran, D., Malinowski, M., Lampinen, A. K., Jaegle, A., McClelland, J. L., Matthey, L., Hill, F., and Lerchner, A. Soda: Bottleneck diffusion models for representation learning, 2023. URL https://arxiv.org/abs/2311.17901.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes, 2022. URL https://arxiv.org/abs/1312.6114.

Lipman, Y., Chen, R. T. Q., Ben-Hamu, H., Nickel, M., and Le, M. Flow matching for generative modeling, 2023. URL https://arxiv.org/abs/2210.02747.

Lipman, Y., Havasi, M., Holderrieth, P., Shaul, N., Le, M., Karrer, B., Chen, R. T. Q., Lopez-Paz, D., Ben-Hamu, H., and Gat, I. Flow matching guide and code, 2024. URL https://arxiv.org/abs/2412.06264.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization, 2019. URL https://arxiv.org/abs/1711.05101.

Ni, Y., Genel, S., Anglés-Alcázar, D., Villaescusa-Navarro, F., Jo, Y., Bird, S., Matteo, T. D., Croft, R., Chen, N., de Santi, N. S. M., Gebhardt, M., Shao, H., Pandey, S., Hernquist, L., and Dave, R. The camels project: Expanding the galaxy formation model space with new astrid and 28-parameter tng and simba suites, 2023. URL https://arxiv.org/abs/2304.02096.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pp. 234–241. Springer, 2015.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. 2021. URL https://arxiv.org/abs/2011.13456.

Villaescusa-Navarro, F., Anglés-Alcázar, D., Genel, S., Spergel, D. N., Li, Y., Wandelt, B., Nicola, A., Thiele, L., Hassan, S., Matilla, J. M. Z., et al. Multifield cosmology with artificial intelligence. *arXiv preprint arXiv:2109.09747*, 2021.

Yang, R. and Mandt, S. Lossy image compression with conditional diffusion models. *Advances in Neural Information Processing Systems*, 36:64971–64995, 2023.

Yue, Z., Wang, J., Sun, Q., Ji, L., Chang, E. I., Zhang, H., et al. Exploring diffusion time-steps for unsupervised representation learning. *arXiv preprint arXiv:2401.11430*, 2024.

A. Architecture and Training Details

In this section, we provide more details on the architecture and training hyperparameters.

A.1. CosmoFlow Model

A.1.1. Encoder

The encoder is a ResNet. Each ResNet block is comprised of [3x3 Conv with circular padding, BatchNorm, 3x3 Conv with circular padding, BatchNorm]. There is a residual connection from the input to the output. The encoder has 6 such ResNet blocks, along with an input convolutional layer (3x3 Conv, BatchNorm, MaxPool), and an output convolution layer (3x3 Conv). The input convolution produces 64 channels. The intermediate ResNet Blocks result in [64, 64, 128, 128, 256, 256] channels, respectively. The output convolution reduces this to 8 channels. This produces the compressed field. We then pass the compressed field through an adaptive average pooling layer to produce the summary statistics.

A.1.2. Decoder

The decoder is a UNet, with 4 downsampling stages and 4 upsampling stages. Each stage consists of four convolutional layers interspersed with batch normalization layers and GeLU activations.

We use sinusoidal positional encoding for the timestep embedding. The summary statistics are passed through a linear layer. These two vectors are used as conditioning for each downsampling and upsampling stage; more precisely, these are passed through an adaptive group normalization operation, then used for channel-wise modulation of the convolutional layer output, as described in [Hudson et al. (2023)]. Self attention modules are used after the second and third downsampling layer, as well as the second upsampling layer.

A.1.3. Training Details

The model is trained for 150 epochs. We use 14,000 samples from the Astrid set for training. The model is trained with the AdamW optimizer [Loshchilov & Hutter (2019)], with the learning rate $\gamma = 0.00005$, $\lambda = 0.01$. We schedule the learning rate to decrease by a factor of 2 whenever the loss plateaus for 10 epochs.

A.2. VAE Model

We compare CosmoFlow to a VAE model with same number of latent dimension. We adopt the encoder and decoder of a VAE image compression model proposed in [(Ballé et al., 2016)]. The encoder is a sequence of 4 downsampling convolution layers with 5x5 kernels, each followed by generalized divisive normalization (GDN) layer [Ballé et al. (2016)] except for the last one. The first three convolution layers each has 128 channels, and the last one has 8 channels, resulting in a encoder output of size $8 \times 16 \times 16$. Two linear layers each with 1024 outputs are used to estimate the mean μ and log-variance $\log(\sigma^2)$ of the 1024 latent variables. The decoder mirrors the encoder in architecture, and is composed of 4 upsampling convolutions each followed by an inverse GDN layer.

The model is trained with batch size of 256 for 300 epochs, using AdamW optimizer with reduced learning rate on plateau.

A.3. Parameter Inference

A.3.1. Parameter Inference on the Summary Statistics

For parameter inference on the summary statistics, we use a fully connected network. We employ Optuna [Akiba et al. (2019)] for hyperparameter optimization. We independently optimize the network hyperparameters for each summary statistics size. For the 8-channel model, we use a single layer network with 2039 neurons. This is trained with the AdamW optimizer, with $\gamma = 1.85 \times 10^{-4}$, $\lambda = 1.09 \times 10^{-7}$, for 200 epochs. However, we observe that the choice of hyperparameters has little effect on the results.

A.3.2. Parameter Inference on the Raw Fields

To do parameter inference on the raw fields, we use a modified version of the ResNet-18 architecture from [He et al. (2016)]. The output of the ResNet is modified to be a 256-dimensional vector, and then a fully-connected layer is added to project it down to 2 dimensions for the output. The model is trained with the AdamW optimizer, with the following parameters:

$\gamma = 0.0002$, $\beta = (0.5, 0.999)$, $\lambda = 0.01$. It is trained for 200 epochs, and we use cosine annealing with $\gamma_{min} = 2 \times 10^{-6}$.

B. Flow Matching Reconstruction Process

In Fig. 6, we demonstrate the inductive bias of flow matching. Images start out as Gaussian noise with flat power spectra. Large-scale features are constructed first, before the small-scale features are added in.

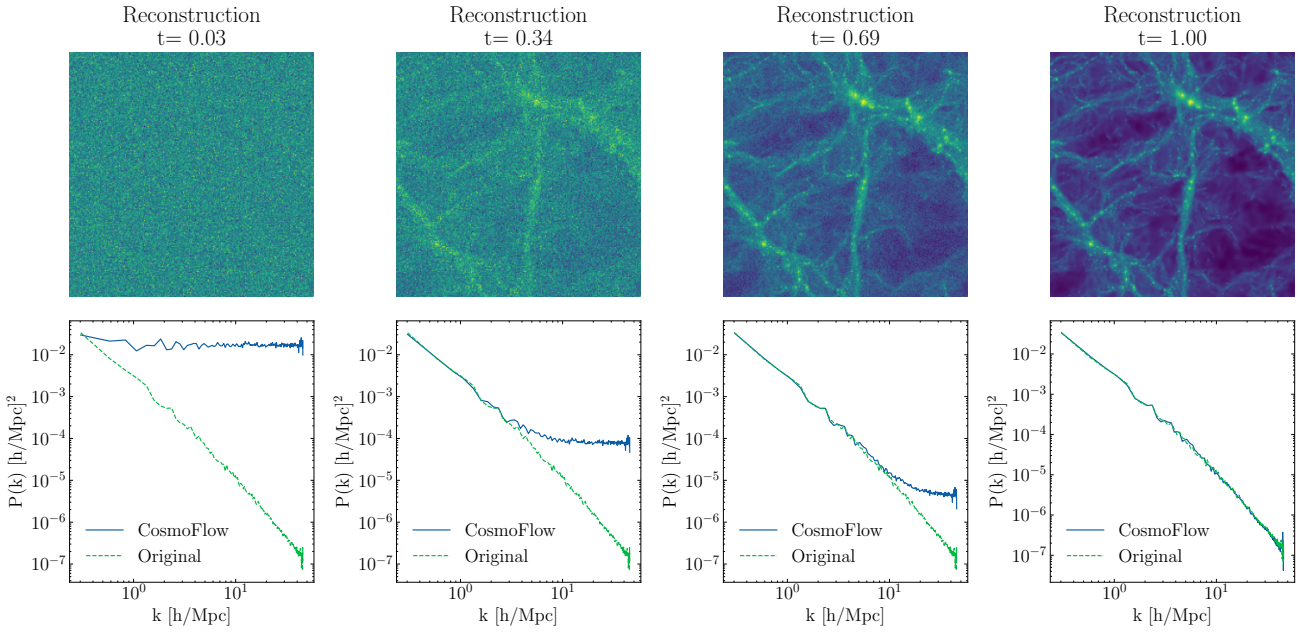

Figure 6. The generation process of a flow matching model. The model starts with a noise sample, and iteratively adds in structure. The power spectrum reflects this, starting as a flat spectrum, before adding in low frequencies and then the high frequencies. Note that for visualization purposes, we normalize the reconstructed power spectrum to match the amplitude of the target at low frequencies at all time steps. This is done by dividing the reconstructed power spectrum by t^2.

C. Interpolation of CDM Fields

In Fig. 7, we show the results of linearly interpolating between two CDM fields from the 1P dataset. These samples have exactly the same initial conditions and cosmological parameters, with the exception of the value of Ω_m. We demonstrate that by linearly interpolating the latent space, we can generate realistic samples at intermediate values of Ω_m, which we cross-validate by showing that our parameter inference network predicts a continuously increasing value for Ω_m over the course of the interpolation.

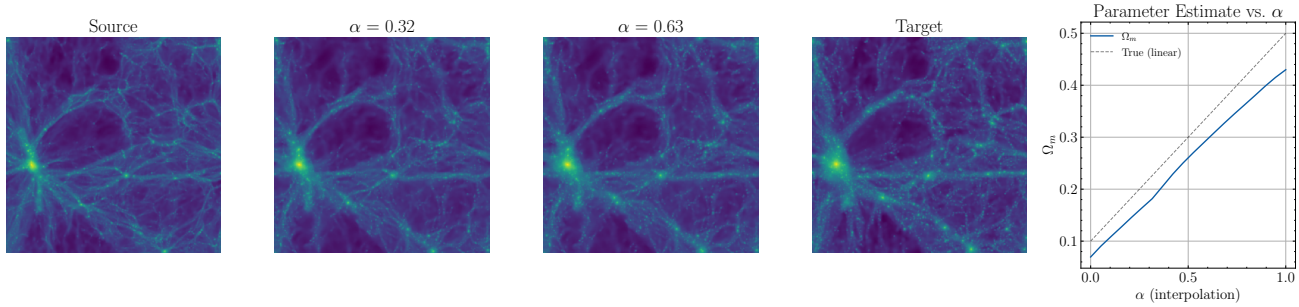

Figure 7. Interpolation between $\Omega_m = 0.1$ to $\Omega_m = 0.5$. Samples remain realistic throughout, and smoothly vary. The parameter inference network also shows a smooth increase in estimated value of Ω_m. Parameter inference was done on the latent representation.