Modality-Agnostic Variational Compression of Implicit Neural Representations

Jonathan Richard Schwarz [*1 2] **Jihoon Tack** [*3] **Yee Whye Teh** [1] **Jaeho Lee** [4] **Jinwoo Shin** [3]

Abstract

We introduce a modality-agnostic neural compression algorithm based on a functional view of data and parameterised as an Implicit Neural Representation (INR). Bridging the gap between latent coding and sparsity, we obtain compact latent representations non-linearly mapped to a soft gating mechanism. This allows the specialisation of a shared INR network to each data item through subnetwork selection. After obtaining a dataset of such latent representations, we directly optimise the rate/distortion trade-off in a modality-agnostic space using neural compression. Variational Compression of Implicit Neural Representations (VC-INR) shows improved performance given the same representational capacity *pre quantisation* while also outperforming previous quantisation schemes used for other INR techniques. Our experiments demonstrate strong results over a large set of diverse modalities using the same algorithm without any modality-specific inductive biases. We show results on images, climate data, 3D shapes and scenes as well as audio and video, introducing VC-INR as the first INR-based method to outperform codecs as well-known and diverse as JPEG 2000, MP3 and AVC/HEVC on their respective modalities.

1. Introduction

Data compression has become a critical problem in the modern era, as vast amounts of data is added to and transmitted through computer networks (Clissa, 2022) at previously unimaginable rates. While momentous progress has been made compared to naive representations, custom compression techniques are still developed for each modality at hand, carefully introducing inductive biases into new algorithms. While being an undoubtedly successful approach, it has limited the transfer of algorithmic ideas between techniques designed for different forms of data. More importantly, in certain engineering or scientific problems, vast amounts of data may be collected for which no generally accepted compression technique may be available (e.g. the AR/VR domain (Yang et al., 2022), point clouds, remote sensing or climate data), inhibiting progress in such fields.

In this paper, we join a recent group of researchers (e.g. Dupont et al., 2021; 2022b; Schwarz & Teh, 2022) in arguing for a paradigm shift: Making modality-agnosticism a key guiding principle, we advocate for a single algorithmic workbench on which methods applicable to any type of data represented by a coordinate and feature space are developed. This would allow research effort to be pooled and any jointly developed model or learning improvement to benefit multiple downstream compression applications at once.

A promising approach towards realising this idea is the use of Implicit Neural Representations (INRs) or Neural Fields (e.g. Tancik et al., 2020; Sitzmann et al., 2020). An INR relies on a functional interpretation of data, specifically as a mapping from coordinates to features (e.g. $(x, y) \rightarrow (r, g, b)$ for images), which is parameterised a neural network. INRs offer various attractive properties, including upsampling to arbitrary resolution (Chen et al., 2021) or a pathway to new approaches to applications such as generative modeling or classification (Dupont et al., 2022a). For our purpose, the most intriguing property of the INR approach is its inherent modality-agnosticism, as any data point can in theory be represented provided it is expressed as a coordinate to feature mapping and thus learnable. Consequently, a learned INR is simply an encoding of the data point within the weights of a neural network, the efficient storage of which has received much attention at a time of ever increasing model capacity. We can thus state the second guiding principle of the work at hand: *Data- as model-compression*.

This second principle distinguishes our ideas from much of the existing work on Neural Compression (e.g. Ballé et al., 2017; 2018; Cheng et al., 2020b), which directly encodes a given data point into a codespace, hence relying on carefully designed modality-specific encoding and decoding networks

[*]Equal contribution [1]DeepMind [2]University College London [3]KAIST [4]POSTECH. Correspondence to: Jonathan Richard Schwarz <schwarzjn@gmail.com>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

(often called analysis/synthesis transforms). Throughout the manuscript, we will highlight how we overcome this limitation while building on rather than replacing the work from this community.

Among the recent work on compression with INRs on the other hand, various ideas for the efficient storage of INRs have been explored. So far, proposed compression and quantisation algorithms are relatively simple (e.g. Uniform Quantisation) or rely on a separate per-signal optimisation process (Strümpler et al., 2022), hence significantly increasing runtime. In addition, much of this work relies on ideas borrowed from Meta-Learning (Finn et al., 2017) to decrease encoding times, which opens up various questions about the best trade-off between compact parameterisation and (Meta-) Learning algorithms. Therefore, despite significant efforts, a substantial gap still exists between INR-based compression and the hand-designed compression methods for certain modalities (e.g. JPEG 2000 for images, MP3 for audio).

In this paper, we improve INR-based compression in a twofold approach (i) We experiment with advanced conditioning techniques resulting in better signal-reconstruction *pre-quantisation* (ii) We overcome limitations of previously used quantisation techniques and introduce a learned quantiser, allowing us to maintain significantly higher reconstruction quality at lower file sizes *post-quantisation*. Both directions of investigation adhere to the guiding principles of modality-agnosticism and the view of data as model compression. This presentation is not accidental, as we can think of the two axes of investigation as orthogonal algorithmic considerations. Indeed, any improvement in (i) increases the upper bound of performance maintained in the quantisation and entropy coding steps in (ii), while any improvement in (ii) reduces the gap between upper bound and actually realised performance.

Contributions:

- *Improved conditioning*: We propose a middle ground between recent sparsity and latent coding approaches to compact representations. The proposed technique introduces a non-linear mapping from a latent codes to a low-rank *soft gating* matrix per layer, selecting a sub-network to represent a data item in an underlying INR. This is shown to learn more efficiently and result in better reconstructions compared to previous approaches. Our interpretation and experimental analysis shines new lights onto related ideas explored in other contexts.
- *Improved compression*: We introduce a learned compressor pre-trained on compact latent codes representing training data. As such latent codes may be extracted from any modality, our proposed compressor operates fully modality-agnostic while making use of the same algorithmic insights previously only applicable to specific modalities.

We verify VC-INR on various data modalities, including image, voxels, scene, climate, audio, and video datasets. Overall, our experimental results demonstrate strong results, consistently outperforming previous INR-based compression methods and improving on popular compression schemes such as MP3 on audio and AVC/HEVC on video clips. In particular, VC-INR achieves a new state-of-the-art results on modality-agnostic compression with INRs, improving the Peak Signal to Noise Ratio (PSNR) on the same bits-per-pixel (bpp) bit rate by 3.3 dB for CIFAR-10 (Krizhevsky et al., 2009), by 2 dB on Kodak[1] (both images), 3.5 dB for ERA5 (climate data) (Hersbach et al., 2019) and 9.5 dB for Librispeech (audio) (Panayotov et al., 2015) respectively. In addition, we outperform MP3 on Librispeech by 5.6 dB and HEVC on Videos by 8.8 dB.

Throughout this paper, we express a given data point $\mathbf{x}$ as a set of coordinates $\mathbf{c} \in \mathcal{C}$ and real-valued features $\mathbf{y} \in \mathcal{Y}$ and its corresponding INR representation as $\boldsymbol{\phi} \in \mathbb{R}^D$. Whenever appropriate, we distinguish between N data points using superscripts, i.e., $\{(\mathbf{x}^i, \boldsymbol{\phi}^i)\}_{i=1}^N$ and individual coordinate/feature pairs using subscripts, i.e. $\mathbf{x}^i := \{(\mathbf{c}_j, \mathbf{y}_j)\}_{i=1}^M$.

2. Related Work

INRs are neural networks approximating the functional mapping from coordinate to feature space. INRs are effective methods for modeling complex continuous signals, such as as 2D images (Chen et al., 2021), 3D scenes (Park et al., 2019), videos (Kim et al., 2022), and are even applicable for modeling discrete data, e.g. graphs (Grattarola & Vandergheynst, 2022). To this end, several architectures have been proposed to capture high-frequency signal details, examples being sinusoidal activations (Sitzmann et al., 2020), positional encodings (Mildenhall et al., 2020), and Fourier features (Tancik et al., 2020).

Neural compression is an end-to-end autoencoder-based lossy compression framework aiming to directly minimise the inherent rate/distortion trade-off. This is based on a transform-coding approach (Goyal, 2001) shown in Figure 1a, where a data item $\mathbf{x}$ is transformed into a latent code $\mathbf{z}$ through an analysis transform g_a. During training, quantisation is simulated through uniform noise ($\mathcal{U}$) resulting in a noisy $\widetilde{\mathbf{z}}$ and a corresponding reconstruction $\widetilde{\mathbf{x}} = g_s(\widetilde{\mathbf{z}})$ through the synthesis transform g_s. At test time, $\mathbf{z}$ is quantised (and entropy coded), resulting in codes and reconstructions $\hat{\mathbf{z}}, \hat{\mathbf{x}}$ respectively. Taking g_a, g_s to be deep neural networks, the neural compression paradigm was introduced in (Ballé et al., 2017; Theis et al., 2017), who make theo-

[1] https://www.kaggle.com/datasets/sherylmehta/kodak-dataset

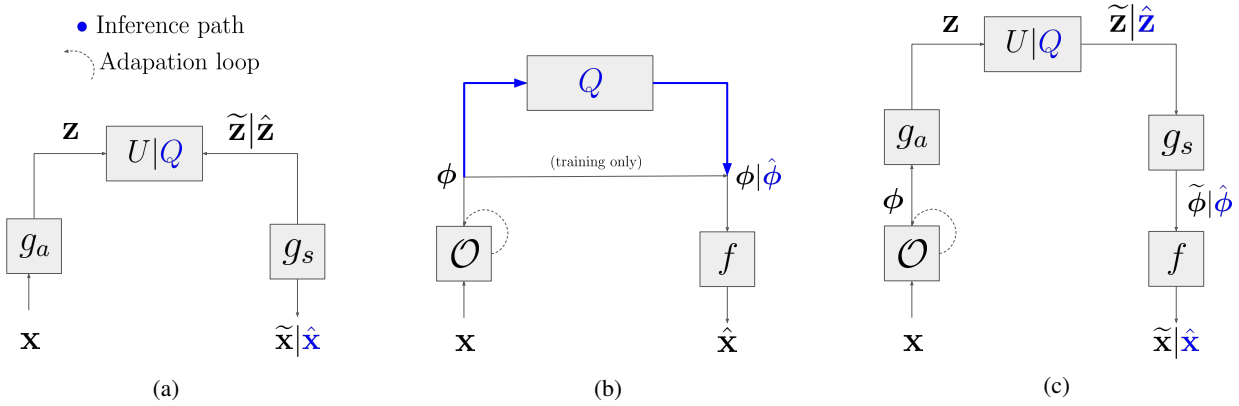


Figure 1. Operational diagrams of learned compression models. Inference time paths are shown in blue. (a) Conventional neural compression (e.g Ballé et al., 2018) (b) Modality-agnostic neural compression with INRs (e.g Dupont et al., 2022b; Schwarz & Teh, 2022) (c) Modality-agnostic variational compression of INRs is built upon the strengths of both techniques. g_a, g_s: Analysis/Synthesis; f: INR network; U/Q: Uniform noise/Quantisation; $\mathcal{O}$: Optimisation process; $\mathbf{x}, \boldsymbol{\phi}, \mathbf{x}$: data point, latent modulation, code element; $\tilde{\mathbf{a}}, \hat{\mathbf{a}}$: Noisy version of $\mathbf{a}$ / Approximation of $\mathbf{a}$. For more details see text.

retical connections to variational inference. Recently, much of the recent work has focused on advanced designs of the entropy model, e.g. by using auto-regressive priors (Minnen et al., 2018a) or various forms of a hierarchical priors (Ballé et al., 2018), such as Gaussian mixture models (GMM) in (Minnen et al., 2018b) and GMMs with attention modules (Cheng et al., 2020b). However, the majority of such neural compression techniques are typically focused on specific modalities, such as images (Lee et al., 2019; Agustsson et al., 2019; Theis et al., 2022) or videos (Lu et al., 2019; Habibian et al., 2019; Agustsson et al., 2020) and feature architectures specifically designed for such modalities, for instance convolutional architectures or the GDN activation function designed for natural images (Ballé et al., 2015).

Data compression with INRs (Figure 1b), introduced by (Dupont et al., 2021) as a modality agnostic compression method required long optimisation processes and architecture search to find a suitable rate/distortion trade-off. Following the wider INR literature (e.g. Tancik et al., 2021), tabula-rasa learning was quickly replaced by a significantly faster Meta-Learning (Finn et al., 2017) adaptation loop (shown as $\mathcal{O}$ in the diagram) while architecture search has been abandoned in favour of compact, instance specific representations $\boldsymbol{\phi}$ on which a deeper, shared INR f is conditioned. The two mainstream approaches have been sparse representations (Lee et al., 2021; Schwarz & Teh, 2022) implementing a close surrogate for the rate loss and/or FiLM-style modulations (Perez et al., 2018; Chan et al., 2021; Mehta et al., 2021) optionally linearly predicted from a compact latent code (Dupont et al., 2022a;b).

Differing from the conventional neural compression workflow, methods following this paradigm either do not feature an explicit quantisation step (Dupont et al., 2021; Lee et al., 2021) beyond default casting to 16-bit representation or rely on simple uniform quantisation based on first and second moment training statistics (Dupont et al., 2022b; Schwarz & Teh, 2022). Recently, Gordon et al. (2023) introduce an alternative quantisation scheme based on K-means clustering, avoiding the likely sub-optimal division of the quantisation space into equally sized regions. Crucially however, subsequent quantisation is not accounted for during training of the previous approaches, forgoing optimisation for deviations in the representations $\hat{\boldsymbol{\phi}}$. This is highlighted by a separate path at inference time in Figure 1b. While advanced quantisation has been introduced (Strümpler et al., 2022), this requires additional training stages, thus increasing encoding runtime. Damodaran et al. (2023) is also similar to one aspect of this work by focusing on improving compression of INRs, showing strong improvements over COIN++ albeit only evaluating the method on images. In terms of applications of compression with INRs, Huang & Hoefler (2022) show the large potential gains in climate applications. Damodaran et al. (2023)

3. Variational Compression of INRs

3.1. Overview

In contrast to the two approaches discussed in the previous section, we now present a computational framework which maintains modality-agnosticism while allowing the use of deep entropy coding. We show a high-level overview in Figure 1c: The method can be best understood as an application of the non-linear transform coding paradigm (Figure 1a) in the compact representation space of the INR approach (Figure 1b).

More concretely, as in other INR techniques, we transform a

data point $\mathbf{x}$ through an adaptation procedure $\mathcal{O}$ into a compact latent representation space $\boldsymbol{\phi}$. We can improve on the relatively simple quantisation techniques in prior works by employing non-linear transforming coding in $\boldsymbol{\phi}$ space, with its analysis and synthesis transforms g_a, g_s now operating on a modality-agnostic representation. This conceptionally simple change has the prime advantage of allowing the use of work from the neural compression literature with minimal changes (limited to simplification of g_a & g_s), thus elevating conventional neural compression to a modality-agnostic paradigm. Compared to prior INR based compression, this allows the direction optimisation of the rate-distortion trade-off (as opposed to using a surrogate) using a deep entropy model. Moreover, a simple forward pass through g_a, subsequent quantisation Q and then g_s is preferable to an iterative technique such as quantisation aware training (Strümpler et al., 2022) at inference time due to runtime considerations.

The rest of this section is split into the two axes of algorithmic improvements presented in this work: After giving a brief description of practical INR-learning on large datasets (Section 3.2), we then (i) Present an improved conditioning technique for specialising the shared base INR f on the data-item specific representation $\boldsymbol{\phi}$ (Section 3.3) (ii) give a detailed discussion of the non-linear transform coding approach use (Section 3.4).

3.2. INRs with Instance-Specific Modulations

An INR is a function $f(\cdot;\boldsymbol{\theta}) : \mathcal{C} \to \mathcal{Y}$ representing a data point through a network with parameters $\boldsymbol{\theta}$. The INR objective is the mean-squared-error of predictions on the data point's coordinates $\{\mathbf{c}_j\}$ and the true features $\{\mathbf{y}_j\}$:

$$\min_{\boldsymbol{\theta}} \sum_{j=1}^{M} \left\| f(\mathbf{c}_j;\boldsymbol{\theta}) - \mathbf{y}_j \right\|_2^2. \tag{1}$$

In practice, naive optimisation would require a large number of iterative steps and result in a set of high-dimensional parameter vectors $\{\boldsymbol{\theta}^i\}$ each representing a data point $\mathbf{x}^i$, making this an unattractive choice.

It is thus attractive to introduce a low dimensional data-item specific parameter $\boldsymbol{\phi}^i$ to model variations in f, while the much larger $\boldsymbol{\theta}$ is used to capture structure across a dataset. The shared INR $f(\cdot;\boldsymbol{\theta})$ is specialised to $\mathbf{x}^i$ through $\boldsymbol{\phi}^i$ resulting in $f(\cdot;\boldsymbol{\theta},\boldsymbol{\phi}^i)$. A reduction in the number of iterative steps per data item is achieved through Meta-Learning (Finn et al., 2017), allowing $\boldsymbol{\phi}^*$ for a test data point $\mathbf{x}^*$ to be obtained in a handful of optimisation steps (see Appendix for details on Meta-Learning).

Common ways to condition f on $\boldsymbol{\phi}^i$ are layer-specific modulations $\mathbf{s}^{(l)}$ obtained by indexing into $\boldsymbol{\phi}^i$, i.e. $\boldsymbol{\phi}^i = [\mathbf{s}^{(1)}, \dots, \mathbf{s}^{(L)}]$ (Mehta et al., 2021). These modulations take the form of FiLM-style (Perez et al., 2018) shifts, i.e. $\mathbf{c}^{(l-1)} \mapsto h(\mathbf{W}^{(l)}\mathbf{c}^{(l-1)} + \mathbf{b}^{(l)} + \mathbf{s}^{(l)})$, where $\mathbf{W}^{(l)}, \mathbf{b}^{(l)}$ are shared weights and biases and h is the activation function. To further reduce the size of $\boldsymbol{\phi}^i$, modulations of an L-layer INR $\mathbf{s} := [\mathbf{s}^{(1)}, \dots, \mathbf{s}^{(L)}]$ can be predicted from $\boldsymbol{\phi}^i$ using a shared linear mapping as $\mathbf{s} = \mathbf{W}^{'}\boldsymbol{\phi} + \mathbf{b}^{'}$ (Dupont et al., 2022a) or alternatively by pruning dimensions in $\boldsymbol{\phi}^i$ through sparsity (Schwarz & Teh, 2022). Both techniques have their own drawbacks: Predictions of $\mathbf{s}$ from $\boldsymbol{\phi}$ have been challenging to train and so far been limited to linear mappings, thus lacking representational capacity. Sparsity techniques on the other hand require approximate inference, introducing additional complexity and various new hyperparameters.

3.3. INR Specialisation through Subnetwork Selection

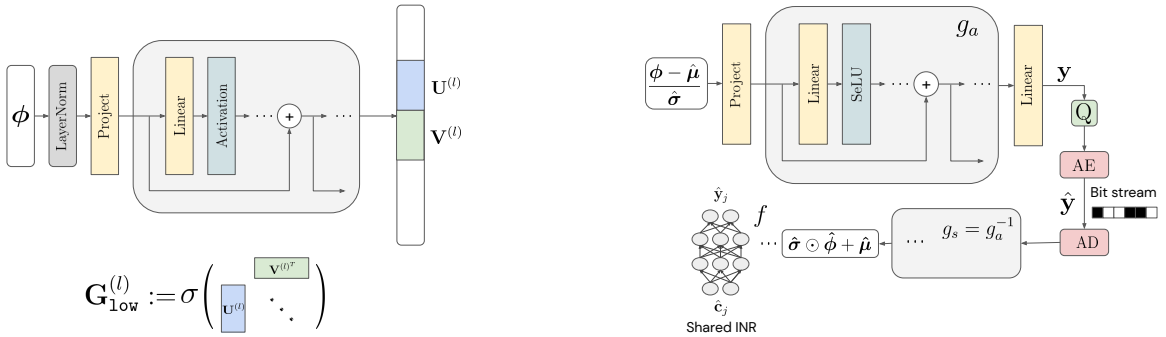
Instead, we take inspiration from both perspectives while overcoming their respective limitations. Following the sparsity paradigm, we observe that while a single network may be conditioned on potentially hundreds of distinct tasks through subnetwork selection (Frankle & Carbin, 2018; Schwarz et al., 2021), it is unclear whether this must be done through hard gating (i.e. requiring *exact* zeros and ones) and thus require approximate inference. Indeed, recent work (He et al., 2019) suggests that soft-gating in the form of the output of a sigmoid $\sigma(x) = \frac{1}{1+e^{-x}}$ may be sufficient. In addition, it is clear that the idea of parametric predictions from $\boldsymbol{\phi}$ may in principle be used in conjunction with subnetwork selection, as a compact $\boldsymbol{\phi}$ could then concentrate its capacity on the non-sparse entries of $\mathbf{s}$, hence naturally combining both ideas.

To this end, we thus suggest the use of a non-linear prediction network mapping $\boldsymbol{\phi}$ to low-rank soft gating masks taking the same shape as the weight-matrices of each layer (see Figure 2a). The functional form of the soft-gating masks is inspired by (Skorokhodov et al., 2021) and takes the form of a low-rank matrix obtained through the outer product of two vectors non-linearly predicted from $\boldsymbol{\phi}$. This choice is sensible for two reasons: First, low-rank parameterisation is widely used as an effective tool for parameter reductions (Phan et al., 2020) and secondly, such modulation have shown potential in representing complex signals such as high-resolution images (Skorokhodov et al., 2021) and videos (Yu et al., 2022). Formally, given the activations of the preceding layer $\mathbf{c}^{(l-1)}$, the transformation of each layer l is

$$\mathbf{c}^{(l-1)} \mapsto \sin(\omega_0(\mathbf{G}^{(l)}_{\texttt{low}} \odot \mathbf{W}^{(l)}\, \mathbf{c}^{(l-1)} + \mathbf{b}^{(l)})) \tag{2}$$

$$\mathbf{G}^{(l)}_{\texttt{low}} := \sigma(\mathbf{U}^{(l)}\mathbf{V}^{(l)\top}), \tag{3}$$

where $\mathbf{U}^{(l)}, \mathbf{V}^{(l)} \in \mathbb{R}^{m\times d}$ are data specific parameters with $d \ll m$, $\odot$ is element-wise multiplication and $\sigma(\cdot)$ the sigmoid operator. Here, we use sinusoidal activation function with its hyperparameters $\omega_0 \in \mathbb{R}^+$ introduced for INRs in (Sitzmann et al., 2020). The **central hypothesis**



(a) Non-linear projection from latent representation ϕ to $\mathbf{G}^{(l)}_{\texttt{low}}$.

(b) Non-linear transform coding in latent representation space.

Figure 2. Architectural details of full model. AE/AD: Arithmetic Encoding/Decoding

of this approach is that $\mathbf{G}^{(l)}_{\texttt{low}}$ acts as a subnetwork selection method, effectively determining and scaling the entries in each weight matrix $\mathbf{W}^{(l)}$ that allow accurate modeling of the data point at hand. We show evidence for this phenomenon in the experimental section.

As before, we can reduce the dimensions of low-rank modulation further, obtaining $[\mathbf{U}^{(1)}, \mathbf{V}^{(1)}, \cdots, \mathbf{U}^{(N)}, \mathbf{V}^{(N)}]$ directly from the compact representation ϕ by predicting a long vector, subsequently reshaped into the respective matrices. Unlike existing methods utilising a linear mapping (Dupont et al., 2022a;b), we use deep residual networks to increase the expressive power, enabled through various stabilisation techniques:

Stabilisation Techniques In line with prior work, we find the direct optimisation of non-linear networks via Meta-Learning to be unstable and under-performing. As low-rank parameterisations are also known to suffer from stability issues, the direct use yield unsatisfactory results. Instead, we suggest three stabilising techniques. (1) First, we propose the normalisation of the modulation ϕ with LayerNorm (Ba et al., 2016), i.e., $\phi \mapsto \texttt{LayerNorm}(\phi)$ as in Fig 2a. Intuitively, this results in higher order gradient optimisation becoming more stable as a normalisation scheme reduces the sharpness of the gradients (Santurkar et al., 2018; Xu et al., 2019). (2) We find residual connections and increasing layer widths (up to computational limits) to aid gradient propagation and significantly increase the performance of non-linear networks. (3) We hypothesise that the sigmoidal bounding of $\mathbf{G}^{(l)}_{\texttt{low}}$ itself has a stabilising effect, preventing the matrix norm from divergence.

At this point it is worth noting that the combination of subnetwork selection techniques and non-linear predictors are not unique to compression and indeed may be beneficial in the wide array of downstream applications made possible through the INR paradigm (Dupont et al., 2022b). Next, we explain the subsequent quantisation of ϕ.

3.4. Variational Compression of Latent Modulations

The key to using non-linear transform coding in a modality-agnostic paradigm is the observation that ϕ may be obtained from data of any kind. For a given modulation ϕ, our goal is now to encode the modulation into a code $\mathbf{z} = g_a(\phi)$ with low Shannon cross-entropy (its rate) and a reconstruction $\hat{\phi} = g_s(\hat{\mathbf{z}})$ with low distortion from ϕ after quantisation $\hat{\mathbf{z}} = Q(\mathbf{z}) = \text{round}(\mathbf{z})$. Because $\hat{\mathbf{z}}$ is discrete, it can be lossless compressed using *entropy coding* such as arithmetic or Huffman coding (Salomon, 2004) to obtain a bit stream.

Here, we use the deep-factorised prior introduced for images in (Ballé et al., 2017) and used as the basis of many follow-up works. The authors establish the interpretation of a relaxed rate-distortion performance as variational autoencoder under a specific generative and inference model, lending the name VC-INR to our method.

We state the compression loss as the sum of (i) the *rate* of the code and (ii) the *distortion* of the recovered signal:

$$\begin{aligned}
&\mathcal{L}_{\texttt{compress}}(\boldsymbol{\pi}_a, \boldsymbol{\pi}_s, \mathbf{x}, \phi) \\
&= \mathcal{L}_{\texttt{rate}} + \lambda \mathcal{L}_{\texttt{distortion}} \\
&= -\log_2[p_{\hat{\mathbf{z}}}(Q(g_a(\phi; \boldsymbol{\pi}_a)))] + \lambda \mathcal{L}_{\texttt{MSE}}(g_s(\hat{\mathbf{z}}; \boldsymbol{\pi}_s), \phi)
\end{aligned} \tag{4}$$

with $p_{\hat{\mathbf{z}}}$ the entropy model, $\mathcal{L}_{\texttt{MSE}}$ the mean squared error (MSE), and $\boldsymbol{\pi}_a, \boldsymbol{\pi}_s$ parameters of the analysis and synthesis transforms. The reconstruction $\hat{\phi}$ is decoded from the quantised code $\hat{\mathbf{z}}$. To optimise this loss, we follow (Ballé et al., 2017) by approximating the discrete quantisation with uniform noise $\mathcal{U}(-\frac{1}{2}, \frac{1}{2})$ to generate a noisy code $\tilde{\mathbf{z}}$ and use the differentiable prior $p_{\tilde{\mathbf{z}}}$ with a non-parametric piecewise linear density model (Ballé et al., 2018).

We show architectural details in Figure 2b: Differing from the typical design of g_a, g_s we do not make use of activations with local gain control and find the SeLU activation (Klambauer et al., 2017) sufficient. In addition, as the vectors ϕ are flat regardless of modality, we can simplify the design of both networks to residual MLPs, removing an-

other form of modality specificity. Finally, we note that the distortion term in Equation (4) is merely a surrogate for the real reconstruction quality of the data $\mathbf{x}$. We thus modify $\mathcal{L}_{\texttt{distortion}}$ to measure distortion on data directly:

$$\mathcal{L}_{\texttt{distortion}} = \mathcal{L}_{\texttt{MSE}}(f(\cdot; \boldsymbol{\theta}, g_s(\hat{\mathbf{z}}; \boldsymbol{\pi}_s)), \mathbf{y}) \tag{5}$$

which we observe to result in the highest quality reconstructions. At this point we emphasise that advanced techniques (Ballé et al., 2018; Cheng et al., 2020a) may be straightforwardly introduced.

4. Experiments

So far, we have discussed a two-fold approach: (i) Advanced conditioning to better capture an underlying signal within a fixed representation *pre-quantisation*. (ii) Variational compression subsequently trained on datasets of such representations. In our empirical evaluation, we will first demonstrate the effectiveness of (i) in isolation (as its results is an upper bound for distortion performance). We then demonstrate the combination of both ideas on a range of compression problems. This will help clearly delineate performance gains as well as provide additional insights into the technique. Throughout the section, we primarily evaluate the performance using the Peak Signal to Noise Ratio (PSNR): $-10 \cdot \log_{10}(\text{MSE})$, where MSE is the mean-squared error between the original and the reconstructed signal.

4.1. Effectiveness of Advanced Conditioning

Table 1. Results for various latent modulation sizes. Shown is voxel accuracy (ShapeNet10) and PSNR (others).

Dataset	Model	Performance @ $\dim(\boldsymbol{\phi})$				
		64	128	256	512	1024
ERA5 (4×)	Functa	43.2	43.7	43.8	44.0	44.1
	MSCN	44.6	45.7	46.0	46.6	46.9
	VC-INR	**45.0**	**46.2**	**47.6**	**49.0**	**50.0**
CelebA-HQ	Functa	21.6	23.5	25.6	28.0	30.7
	MSCN	21.8	23.8	25.7	28.1	**30.9**
	VC-INR	**22.0**	**23.9**	**26.0**	**28.3**	30.8
SRN Cars	Functa	22.4	23.0	23.1	23.2	23.1
	MSCN	22.8	**24.0**	**24.3**	24.5	24.8
	VC-INR	**23.9**	**24.0**	**24.3**	**25.2**	**25.5**
ShapeNet10	Functa	$_{99}.30$	$_{99}.40$	$_{99}.44$	$_{99}.50$	$_{99}.55$
	MSCN	$_{99}.43$	$_{99}.50$	$_{99}.56$	$_{99}.63$	$_{99}.69$
	VC-INR	$_{99}\mathbf{.54}$	$_{99}\mathbf{.61}$	$_{99}\mathbf{.64}$	$_{99}\mathbf{.70}$	$_{99}\mathbf{.71}$

We first evaluate our method pre-quantisation on various modalities including images using CelebA-HQ dataset (Karras et al., 2018), manifolds using ERA5 (Hersbach et al., 2019), 3d NeRF scenes using the SRN cars (Sitzmann et al., 2019) and 3d voxels using the top 10 classes of ShapeNet (Chang et al., 2015). Following prior work, we train SIREN with 15 layers of 512 units and use MetaSGD (Li et al., 2017) with 3 inner-loop steps as our Meta-Learning method and use the same task batch size for comparable conditions. For baselines, we compare our technique with latent modulations using Functa (Dupont et al., 2022a) and sparse modulations using MSCN (Schwarz & Teh, 2022). More details in the Appendix.

As illustrated in Table 1, we demonstrates a marked improvement over previous approaches in almost all cases. Particularly noteworthy, VC-INR outperforms MSCN on ERA5 by more than 3.1dB when using a modulation size of 1024. This is particularly significant, as PSNR is based on a logarithmic scale. In addition, we note that the use of more complex latent $\rightarrow$ modulations/mask networks (as opposed to the linear projection of Functa) not only leads to better results, but also exhibits significantly faster learning progress (Figure 3a).

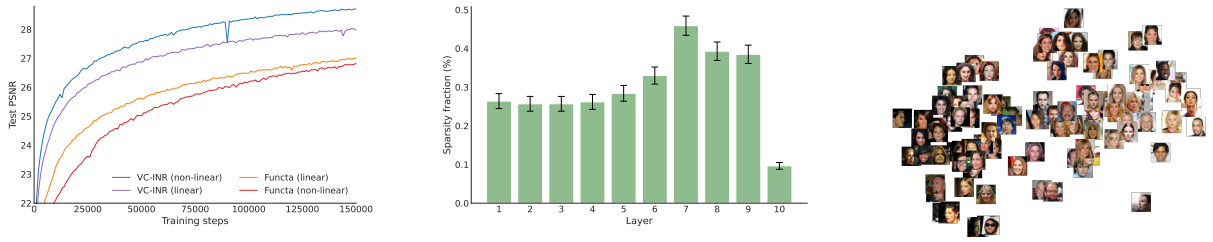
A key hypothesis of our proposed conditioning technique is the idea of subnetwork selection. To provide empirical evidence and understand the behaviour of our conditioning method, we analyse the masks $\mathbf{G}^{(l)}_{\texttt{low}}$ after obtaining $\boldsymbol{\phi}^i$ for test set images. This is shown in Figure 3. First, we note that product of gating masks and a shared, Meta-Learning matrix does indeed implement moderate sparsity levels (which we define as $|(\mathbf{G}_{\texttt{low}} \odot \mathbf{W})_{ij}| < 0.001$), despite avoiding the use of approximate inference (Figure 3b). Remarkably, we observe sparsity levels varying significantly per layer, suggesting VC-INR learns *where to learn*. This is particular significant as it is well known that only a fraction of layers typically need to adapted in Meta-Learning (Zintgraf et al., 2019). Indeed, this was a key insight of MSCN (Schwarz & Teh, 2022) which we share despite our use of a much simpler sparsity/gating method.

Moreover, further examining our soft gating mechanism, we provide a t-SNE visualization (Maaten & Hinton, 2008) of the adapted masks on CelebA-HQ (Figure 3c). Resulting patterns intriguingly show clear clustering according to image characteristics such as background color, indicating the ability to condition the shared INR based on image statistics.

4.2. Data Compression Across Modalities

We now evaluate VC-INR for data compression, the primary focus of our work. To demonstrate the versatility of VC-INR, we examine a range frequently encountered modalities. We measure reconstruction performance measured in terms of PSNR under different levels of compressed data sizes measured in kilobits per second (kbps) for audio and bits-per-pixel (bpp)[2]. Baselines are codecs such as JPEG (Wallace, 1992), JPEG 2000 (Skodras et al., 2001), BPG

[2] $\text{bpp} = \frac{\text{bits per parameters} \times \text{number of parameters}}{\text{number of pixels}}$



(a) Learning curves (b) Sparsity pattern (c) Mask clustering on CelebA-HQ

Figure 3. Analysis of the VC-INR on CelebA-HQ (a) Learning curves of Functa, linear & non-linear VC-INR models (b) Sparsity patterns of adapted weights throughout the network (c) t-SNE (Van der Maaten & Hinton, 2008) visualisation of masks $\mathbf{G}_{\texttt{low}}$ after adaptation.

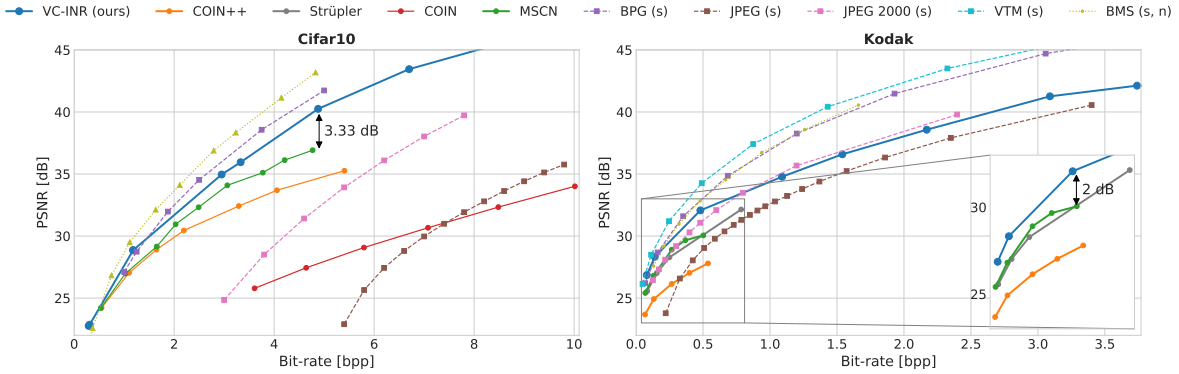


Figure 4. Compression results on image datasets CIFAR-10 (left) & Kodak (right). Modality-specific approaches are shown with a dashed line and marked *(s)*. Conventional neural compression methods (also modality-agnostic) with a dotted line and marked *(s, n)*. BMS is (Ballé et al., 2017), Strüpler is (Strümpler et al., 2022) and VTM (Bross et al., 2021).

(Bellard, 2014), MP3 (MP3, 1993), AVC (Wiegand et al., 2003), and HEVC (Bross et al., 2021). We also compare against the modality-specific neural compression scheme BMS (Ballé et al., 2018), VTM (Bross et al., 2021) and other INR techniques such as COIN (Dupont et al., 2021), COIN++ (Dupont et al., 2022b), MSCN (Schwarz & Teh, 2022) and the method in (Strümpler et al., 2022).

Uniform vs Variational Compression

While the previous section provides empirical justification for the architectural changes of VC-INR, we now additionally show the effectiveness of the proposed quantisation method. Figure 6 contrasts rate-distortion curves obtained using Uniform Quantisation with the transform coding setup introduced earlier. Results are obtained by compressing the latent vectors obtained from two pre-trained models to varying bit-rates by varying the number of bits for uniform quantisation or the rate-distortion trade-off parameter λ for the full VC-INR model. We note that the use of Ballé et al. (2017)'s transform coding drastically shifts the rate-distortion curves towards lower bit-rates while maintaining a better reconstruction ratio. Furthermore, we can also see that the improved pre-training effectively increases the ceiling reconstruction performance when comparing uniform quantisation for VC-INR (light blue) with our COIN++ implementation (orange).

Images We show compression performance on the image domain using the CIFAR-10 (Krizhevsky et al., 2009) and Kodak (meta-trained on Div2k (Agustsson & Timofte, 2017)) datasets. In order to handle the large images found in the Kodak dataset, we divide the images into smaller patches as previously established in prior work.

Figure 4 shows that VC-INR significantly and consistently outperforms prior INR-based data compression methods (COIN, COIN++, MSCN, Strümpler) and even certain image codecs (JPEG/JPEG 2000) on the CIFAR-10 dataset. In addition, VC-INR reconstruction continue to improve with higher bitrates, which we demonstrate by almost pixel-perfect reconstruction. This implies that learned entropy coding is a key factor in achieving strong results. Furthermore, VC-INR shows comparable performance to the strongest modality-specific methods at low bitrates, despite not taking advantage of inductive biases. While not fully matching state-of-the-art (SOTA) results on images compared to all compression techniques, we significantly reduce the gap. Note that we provide further results on Kodak using Multiscale structural similarity index measure (MS-SSIM) in the Appendix.

(a) Compared with COIN++ (Dupont et al., 2022b).

(b) Compared with MSCN (Schwarz & Teh, 2022).

Figure 5. Qualitative results from the Kodak dataset. Shown are VC-INR models in comparison with other INR-based techniques at similar bit-rates (3rd column) as well as a high-quality model (last column).

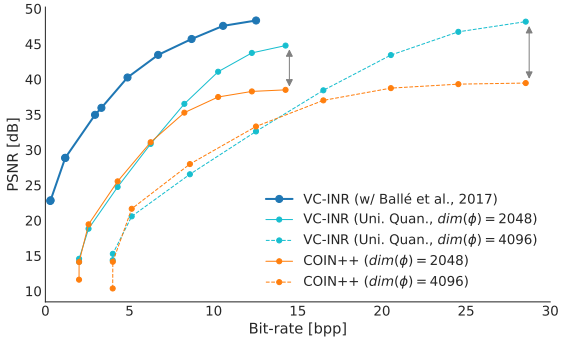


Figure 6. Learned vs uniform quantisation for VC-INR & COIN++ on CIFAR-10.

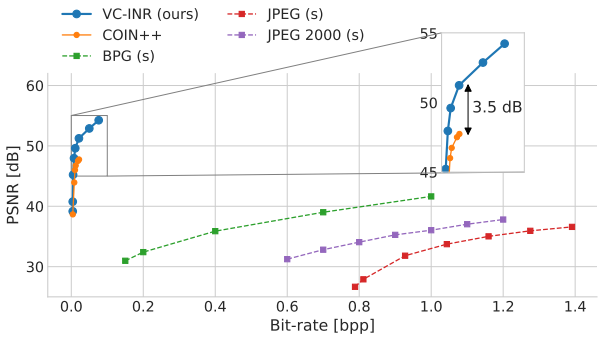


Figure 7. Compression results on ERA5 (climate data/manifolds).

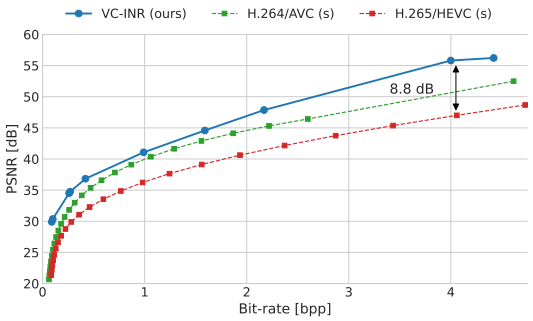
Manifolds This evaluates VC-INR on global temperature measurements from the ERA5 (16×) dataset. The dataset consists of temperature measurements (features) at equally spaced latitudes and longitudes (coordinates) on Earth from 1979 to 2020, represented by spherical coordinates. Since no codec or neural compression method has been developed specifically for this modality, we compare VC-INR to COIN++ and image codecs (applied by unrolling the manifold on a rectangular grid) as baselines. As shown in Figure 7, VC-INR, we achieve an improvement of approximately 3.5dB at the same bitrate compared to the SOTA. This highlights the large potential impact modality-agnostic techniques might have for specialised data types.

Audios Evaluating VC-INR in the audio domain, we utilise the LibriSpeech dataset (Panayotov et al., 2015), a large speech dataset recorded at a 16kHz sampling rate. We consider the MP3 codec as well as COIN++ as baseline methods. As in COIN++ we use patching to keep the comparison fair. Our results, shown in Figure 8b demonstrate impressive results, showing that VC-INR significantly outperforms both COIN++ as well as the widely used and popular MP3 codec.

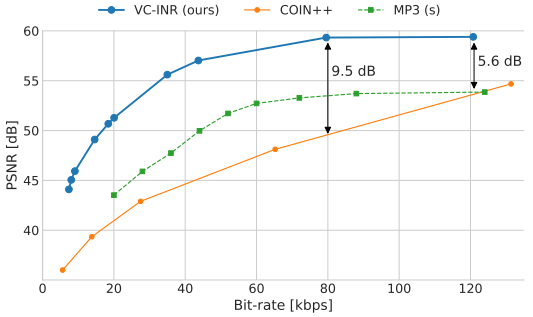
Videos Turning to the video domain, we compress clips from the UCF-101 action recognition dataset (Soomro et al., 2012), once again using patching. Here, we compare VC-INR to video codecs AVC and HEVC. Impressively, VC-INR outperforms both, raising hopes for the potential of INR-based compression to one day replace hand-designed codecs for video. Qualitative results are available in Figure 9, showing the prediction errors of VC-INR models at varying bit-rates, for all of which we achieve SOTA better results.

5. Conclusion

We introduce VC-INR, a modality-agnostic neural compression technique showing strong and consistent improvements over previous INR-based methods. This was achieved by developing algorithmic improvements across the two axes of representational power and advanced quantisation while maintaining modality-agnosticism. Our technique bridges the gap between recent approaches to compact INR representations based on latent codes and sparsity and shows how previously modality-specific algorithms can be elevated to the modality-agnostic setting. Our evaluation shows strong improvement on previous work with INRs (e.g. Dupont et al., 2022b; Schwarz & Teh, 2022) while outperforming certain established algorithms (e.g. JPEG on images, MP3 on audio and AVC/HEVC on videos) while reducing the gap to others (e.g. BPG or BMS on images). We believe that the conceptual advantage of a single algorithm applicable to all modulations will continue to show rapid improvements



(a) UCF-101



(b) LibriSpeech

Figure 8. Compression results on (a) videos and (b) audio.

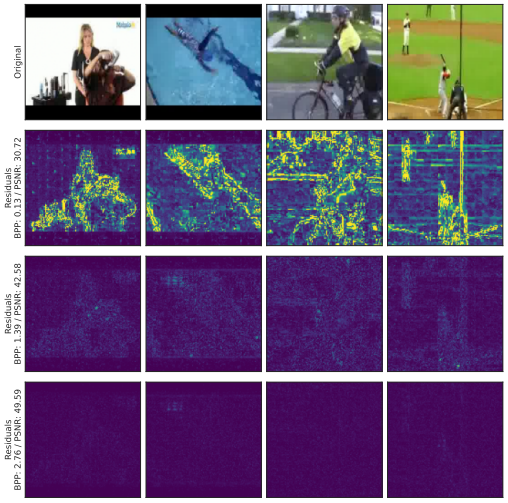


Figure 9. Results on videos showing residuals of VC-INR at various quality levels. Videos available: 0.13 bpp, 1.39 bpp, 2.76 bpp.

as innovations are developed. Future work may focus on including further developments such as advanced priors (e.g. Ballé et al., 2018; Cheng et al., 2020b; Ladune et al., 2022). In addition, improved patching strategies resulting from e.g. memory-efficient Meta-Learning algorithms or path allocation based on signal variation might be fruitful.

Acknowledgements

We thank Hyunjik Kim and Danilo J. Rezende for their insightful comments and feedback. We also thank Strümpler et al. (2022) for sharing their results. This work is supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-00075, Artificial Intelligence Graduate School Program (KAIST), No.2021-0-02068, Artificial Intelligence Innovation Hub, and No.2022-0-00713, Meta-learning applicable to real-world problems) and National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (2022R1F1A1075067).

References

Agustsson, E. and Timofte, R. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017.

Agustsson, E., Tschannen, M., Mentzer, F., Timofte, R., and Gool, L. V. Generative adversarial networks for extreme learned image compression. In *IEEE International Conference on Computer Vision*, 2019.

Agustsson, E., Minnen, D., Johnston, N., Balle, J., Hwang, S. J., and Toderici, G. Scale-space flow for end-to-end optimized video compression. In *CVPR*, 2020.

Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Ballé, J., Laparra, V., and Simoncelli, E. P. Density modeling of images using a generalized normalization transformation. *arXiv preprint arXiv:1511.06281*, 2015.

Ballé, J., Laparra, V., and Simoncelli, E. P. End-to-end optimized image compression. In *International Conference on Learning Representations*, 2017.

Ballé, J., Minnen, D., Singh, S., Hwang, S. J., and Johnston, N. Variational image compression with a scale hyperprior. In *International Conference on Learning Representations*, 2018.

Bellard, F. Bpg image format. https://bellard.org/bpg/, 2014.

Bross, B., Wang, Y.-K., Ye, Y., Liu, S., Chen, J., Sullivan, G. J., and Ohm, J.-R. Overview of the versatile video coding (vvc) standard and its applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

Chan, E. R., Monteiro, M., Kellnhofer, P., Wu, J., and Wetzstein, G. pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2021.

Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

Chen, Y., Liu, S., and Wang, X. Learning continuous image representation with local implicit image function. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2021.

Cheng, Z., Sun, H., Takeuchi, M., and Katto, J. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020a.

Cheng, Z., Sun, H., Takeuchi, M., and Katto, J. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020b.

Clissa, L. Survey of big data sizes in 2021. *arXiv preprint arXiv:2202.07659*, 2022.

Damodaran, B. B., Balcilar, M., Galpin, F., and Hellier, P. Rqat-inr: Improved implicit neural image compression. In *2023 Data Compression Conference (DCC)*, pp. 208–217. IEEE, 2023.

Dupont, E., Goliński, A., Alizadeh, M., Teh, Y. W., and Doucet, A. Coin: Compression with implicit neural representations. In *ICLR Neural Compression: From Information Theory to Applications Workshop*, 2021.

Dupont, E., Kim, H., Eslami, S., Rezende, D., and Rosenbaum, D. From data to functa: Your data point is a function and you should treat it like one. In *International Conference on Machine Learning*, 2022a.

Dupont, E., Loya, H., Alizadeh, M., Goliński, A., Teh, Y. W., and Doucet, A. Coin++: Data agnostic neural compression. *Transactions on Machine Learning Research*, 2022b.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, 2017.

Frankle, J. and Carbin, M. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

Gordon, C., Chng, S.-F., MacDonald, L., and Lucey, S. On quantizing implicit neural representations. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 341–350, 2023.

Goyal, V. K. Theoretical foundations of transform coding. *IEEE Signal Processing Magazine*, 2001.

Grattarola, D. and Vandergheynst, P. Generalised implicit neural representations. In *Advances in Neural Information Processing Systems*, 2022.

Habibian, A., Rozendaal, T. v., Tomczak, J. M., and Cohen, T. S. Video compression with rate-distortion autoencoders. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

He, X., Sygnowski, J., Galashov, A., Rusu, A. A., Teh, Y. W., and Pascanu, R. Task agnostic continual learning via meta learning. *arXiv preprint arXiv:1906.05201*, 2019.

Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Rozum, I., et al. Era5 monthly averaged data on single levels from 1979 to present. *Copernicus Climate Change Service (C3S) Climate Data Store (CDS)*, 2019.

Huang, L. and Hoefler, T. Compressing multidimensional weather and climate data into neural networks. *arXiv preprint arXiv:2210.12538*, 2022.

Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.

Kim, S., Yu, S., Lee, J., and Shin, J. Scalable neural video representations with learnable positional features. In *Advances in Neural Information Processing Systems*, 2022.

Klambauer, G., Unterthiner, T., Mayr, A., and Hochreiter, S. Self-normalizing neural networks. *Advances in neural information processing systems*, 30, 2017.

Krizhevsky, A. et al. Learning multiple layers of features from tiny images, 2009.

Ladune, T., Philippe, P., Henry, F., and Clare, G. Coolchic: Coordinate-based low complexity hierarchical image codec. *arXiv preprint arXiv:2212.05458*, 2022.

Lee, J., Cho, S., and Beack, S.-K. Context-adaptive entropy model for end-to-end optimized image compression. In *International Conference on Learning Representations*, 2019.

Lee, J., Tack, J., Lee, N., and Shin, J. Meta-learning sparse implicit neural representations. In *Advances in Neural Information Processing Systems*, 2021.

Li, Z., Zhou, F., Chen, F., and Li, H. Meta-sgd: Learning to learn quickly for few-shot learning. *arXiv preprint arXiv:1707.09835*, 2017.

Lu, G., Ouyang, W., Xu, D., Zhang, X., Cai, C., and Gao, Z. Dvc: An end-to-end deep video compression framework. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

Maaten, L. v. d. and Hinton, G. Visualizing data using t-sne. *Journal of Machine Learning Research*, 2008.

Mehta, I., Gharbi, M., Barnes, C., Shechtman, E., Ramamoorthi, R., and Chandraker, M. Modulated periodic activations for generalizable local functional representations. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2021.

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamorthi, R., and Ng, R. NeRF: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, 2020.

Minnen, D., Ballé, J., and Toderici, G. Joint autoregressive and hierarchical priors for learned image compression. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, 2018a.

Minnen, D., Ballé, J., and Toderici, G. D. Joint autoregressive and hierarchical priors for learned image compression. In *Advances in Neural Information Processing Systems*, 2018b.

MP3. MP3 codec. https://www.iso.org/standard/22412.html, 1993.

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. Librispeech: an asr corpus based on public domain audio books. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015.

Park, J. J., Florence, P., Straub, J., Newcombe, R., and Lovegrove, S. Deepsdf: Learning continuous signed distance functions for shape representation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

Perez, E., Strub, F., De Vries, H., Dumoulin, V., and Courville, A. Film: Visual reasoning with a general conditioning layer. In *AAAI Conference on Artificial Intelligence*, 2018.

Phan, A.-H., Sobolev, K., Sozykin, K., Ermilov, D., Gusak, J., Tichavskỳ, P., Glukhov, V., Oseledets, I., and Cichocki, A. Stable low-rank tensor decomposition for compression of convolutional neural network. In *European Conference on Computer Vision*, 2020.

Salomon, D. *Data compression: the complete reference*. Springer Science & Business Media, 2004.

Santurkar, S., Tsipras, D., Ilyas, A., and Madry, A. How does batch normalization help optimization? In *Advances in Neural Information Processing Systems*, 2018.

Schwarz, J., Jayakumar, S., Pascanu, R., Latham, P. E., and Teh, Y. Powerpropagation: A sparsity inducing weight reparameterisation. *Advances in Neural Information Processing Systems*, 34:28889–28903, 2021.

Schwarz, J. R. and Teh, Y. W. Meta-learning sparse compression networks. *Transactions on Machine Learning Research*, 2022.

Sitzmann, V., Zollhöfer, M., and Wetzstein, G. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In *Advances in Neural Information Processing Systems*, 2019.

Sitzmann, V., Martel, J. N. P., Bergman, A. W., Lindell, D. B., and Wetzstein, G. Implicit neural representations with periodic activation functions. In *Advances in Neural Information Processing Systems*, 2020.

Skodras, A., Christopoulos, C., and Ebrahimi, T. The jpeg 2000 still image compression standard. *IEEE Signal Processing Magazine*, 2001.

Skorokhodov, I., Ignatyev, S., and Elhoseiny, M. Adversarial generation of continuous images. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2021.

Soomro, K., Zamir, A. R., and Shah, M. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

Strümpler, Y., Postels, J., Yang, R., Van Gool, L., and Tombari, F. Implicit neural representations for image compression. In *European Conference on Computer Vision*, 2022.

Tancik, M., Srinivasan, P. P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J. T., and Ng, R. Fourier features let networks learn high frequency functions in low dimensional domains. In *Advances in Neural Information Processing Systems*, 2020.

Tancik, M., Mildenhall, B., Wang, T., Schmidt, D., Srinivasan, P. P., Barron, J. T., and Ng, R. Learned initializations for optimizing coordinate-based neural representations. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2021.

Theis, L., Shi, W., Cunningham, A., and Huszár, F. Lossy image compression with compressive autoencoders. *arXiv preprint arXiv:1703.00395*, 2017.

Theis, L., Salimans, T., Hoffman, M. D., and Mentzer, F. Lossy compression with gaussian diffusion. *arXiv preprint arXiv:2206.08889*, 2022.

Van der Maaten, L. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Wallace, G. K. The jpeg still picture compression standard. *IEEE Transactions on Consumer Electronics*, 1992.

Wang, Z., Simoncelli, E. P., and Bovik, A. C. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pp. 1398–1402. Ieee, 2003.

Wiegand, T., Sullivan, G. J., Bjontegaard, G., and Luthra, A. Overview of the h. 264/avc video coding standard. *IEEE Transactions on circuits and systems for video technology*, 2003.

Xu, B., Wang, N., Chen, T., and Li, M. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015.

Xu, J., Sun, X., Zhang, Z., Zhao, G., and Lin, J. Understanding and improving layer normalization. In *Advances in Neural Information Processing Systems*, 2019.

Yang, Y., Mandt, S., and Theis, L. An introduction to neural data compression. *arXiv preprint arXiv:2202.06533*, 2022.

Yu, S., Tack, J., Mo, S., Kim, H., Kim, J., Ha, J.-W., and Shin, J. Generating videos with dynamics-aware implicit generative adversarial networks. In *International Conference on Learning Representations*, 2022.

Zintgraf, L., Shiarli, K., Kurin, V., Hofmann, K., and Whiteson, S. Fast context adaptation via meta-learning. In *International Conference on Machine Learning*, 2019.

A. Dataset Description

CelebA-HQ is a high-quality version of the CelebA dataset, which includes images of celebrities along with corresponding attributes (Karras et al., 2018). By following (Dupont et al., 2022a), we divide the dataset into 27,000 training examples and 3,000 test examples, and pre-processed the pixel coordinates into $[0, 1]^2$ and feature values ranging from 0 to 1.

ShapeNet is a dataset of 3D shapes of 10 different object categories (Chang et al., 2015). We follow the pre-processing by (Dupont et al., 2022a), and downscale the resolution of 128^3 to 64^3 by using `scipy.ndimage.zoom` function with threshold 0.05. To augment the datasets, the authors applied a 50-fold expansion by independently scaling the shapes in the x, y, and z axes using a randomly sampled scale within the range of 0.75 to 1.25. The resulting dataset includes 1,516,750 training examples and 168,850 test examples with voxel coordinates into $[0, 1]^3$ and occupancies in binary $\{0, 1\}$.

ERA5 is a dataset consists of temperature observations from 1979 to 2020 on a global grid of equally spaced latitudes and longitudes (Hersbach et al., 2019). By following (Dupont et al., 2022a), we downsample the grid resolution 721×1044 to 181×360. Each time step is treated as a separate data point, and the dataset is split into a training set of 9676 data points and a test set of 2420 data points. As for the input, the given latitudes $c_{\texttt{lat}}$ and longitudes $c_{\texttt{long}}$ are transformed into 3D Cartesian coordinates $\mathbf{c} = (\cos c_{\texttt{lat}} \cos c_{\texttt{long}}, \cos c_{\texttt{lat}} \sin c_{\texttt{long}}, \sin c_{\texttt{lat}})$ where latitudes $c_{\texttt{lat}}$ are equally spaced in $[-\frac{\pi}{2}, \frac{\pi}{2}]$ and longitudes $c_{\texttt{long}}$ are equally spaced in $[0, \frac{2\pi(n-1)}{n}]$ where n the number of distinct values of longitude (360).

SRN Cars is a dataset of car scenes, with 2458 examples in the training set and 703 examples in the test set (Sitzmann et al., 2019). Each example consists of 50 random views centered on the car in the training set, and 251 views in the test set. The pre-processing of the data was conducted according to the guidelines provided by (Dupont et al., 2022b).

CIFAR-10 is a dataset of 50,000 train and 10,000 test images with a resolution of 32 x 32, comprising 10 different object categories (Krizhevsky et al., 2009). We use the same pre-processing as in CelebA-HQ dataset.

Kodak is a dataset of 24 uncompressed PNG images with a resolution of 768×512, provided by the Kodak corporation. By following (Schwarz & Teh, 2022), we meta-learn on the high-quality versions of the Div2K dataset (Agustsson & Timofte, 2017), which consists of 900 images (by combining train and validation set). For Meta-Learning, we also train the model on randomly cropped 32×32 patches and for evaluation, we split the image into non-overlapping patches where each modulations are adapted on each patches. Here, we also use the same pre-processing as in CelebA-HQ dataset.

LibriSpeech is a collection of read English speech recordings at a 16kHz sampling rate (Panayotov et al., 2015). By following Dupont et al. (2022b), we use the train-clean-100 split, which consists of 28,539 examples, and the test-clean split, which consists 2,620 examples. For the experiments, we use the first 3 seconds of each example (which is 48,000 audio samples) for both training and evaluation. For the pre-processing, we scale the coordinates into $[-5, 5]$.

UCF-101 is a video action dataset comprising 13,320 videos with a resolution of 320×240, organised into 101 classes (Soomro et al., 2012). In order to standardise the input for the model, we center-crop each video clip to $240 \times 240 \times 24$ and then resized to $128 \times 128 \times 24$.

B. Numerical Results

For the sake of reputability, we now state the numerical compression values used to plot the results in Section 4. Note that baseline results have been taken from the code repository for COIN++ (Dupont et al., 2022b):

```
# Cifar-10
vcinr_bpp = [0.29, 0.31, 1.18, 1.18, 2.95, 3.33, 4.88, 6.70, 8.69, 10.56, 12.52]
vcinr_psnr = [22.76, 22.86, 28.86, 28.86, 34.96, 35.95, 40.25, 43.45,  45.70, 47.56, 48.32]

# Kodak
vcinr_bpp = [0.08, 0.14, 0.48, 1.09, 1.54, 2.17, 3.09, 3.74, 5.56]
vcinr_psnr = [26.86, 28.33, 32.07, 34.78, 36.59, 38.57, 41.26, 42.12, 42.24]

# ERA-5
vcinr_bpp = [0.004, 0.004, 0.005, 0.00758, 0.011, 0.02119, 0.05, 0.07616]
vcinr_psnr = [39.172, 40.766, 45.219, 47.965, 49.612, 51.25, 52.89, 54.25]

# Librispeech
vcinr_bpp = [7.38, 8.04, 9.06, 14.61, 18.42, 20.06, 34.99, 43.69, 79.54, 120.77]
```

```
vcinr_psnr = [44.10, 45.05, 45.93, 49.10, 50.68, 51.28, 55.61, 57.03, 59.33, 59.40]

#  UCF-101
vcinr_bpp = [0.09, 0.10, 0.26, 0.27, 0.42, 0.99, 1.59, 2.17, 4.00, 4.42]
vcinr_psnr = [29.90, 30.37, 34.51, 34.75, 36.83, 41.07, 44.58, 47.86, 55.81, 56.22]
```

C. Meta-Learning Implicit Neural Representations with Latent Modulations

In order to efficiently and effectively encode a given signal into a compact latent representation, we utilise a Gradient-based Meta-Learning approach, such as model-agnostic meta-learning (MAML) (Finn et al., 2017). In our case, MAML aims to find a good initialisation $\boldsymbol{\phi}_0$ and shared INR parameter $\boldsymbol{\theta}$, allowing for the encoding of a given signal $\mathbf{x}$ into the modulation $\boldsymbol{\phi}$ within a few gradient steps from $\boldsymbol{\phi}_0$. Writing $\mathcal{L}_{\texttt{MSE}}(\boldsymbol{\theta}, \boldsymbol{\phi}, \mathbf{x})$ as a shorthand for the INR fitting loss (Equation (1)), a single gradient step adaptation of MAML is computed as:

$$\boldsymbol{\phi} = \boldsymbol{\phi}_0 - \alpha \nabla_{\boldsymbol{\phi}_0} \mathcal{L}_{\texttt{MSE}}(\boldsymbol{\theta}, \boldsymbol{\phi}_0, \mathbf{x}), \tag{6}$$

where α is the step size used in the inner loop. Note that one can easily iterate the adaptation for multiple steps. The key idea of MAML is to backpropagate through this optimisation process, directly learning an initialisation $\boldsymbol{\phi}_0$ (along with additional shared parameters $\boldsymbol{\theta}$) such that $\boldsymbol{\phi}$ can parameterise a good reconstruction of the signal after adaptation. This is typically computed over the training signal distribution $p(\mathbf{x})$:

$$\min_{\boldsymbol{\theta}, \boldsymbol{\phi}_0} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \Big[\mathcal{L}_{\texttt{MSE}}(\boldsymbol{\theta}, \boldsymbol{\phi}, \mathbf{x}) \Big] = \min_{\boldsymbol{\theta}, \boldsymbol{\phi}_0} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \Big[\mathcal{L}_{\texttt{MSE}}\big(\boldsymbol{\theta}, \boldsymbol{\phi}_0 - \alpha \nabla_{\boldsymbol{\phi}_0} \mathcal{L}_{\texttt{MSE}}(\boldsymbol{\theta}, \boldsymbol{\phi}_0, \mathbf{x}), \mathbf{x}\big) \Big]. \tag{7}$$

Here, we refer each optimisation of MAML, Equation (6) as "inner-loop", and (7) as "outer-loop", respectively. In practise, we also meta-learn the step size α as another parameter updated in the outer loop, an approach known as MetaSGD (Li et al., 2017). This can be interpreted as a pre-conditioning of the gradient.

Algorithm 1 INR Meta-training stage

Data: Dataset $\{\mathbf{x}^i, \mathbf{y}^i\}_{i=1}^N$

Initialise shared network $\boldsymbol{\theta}$ and latent modulation initialisation $\boldsymbol{\phi}_0$.

while *not converged* **do**

  Sample batch of data $\mathcal{B} = \{\mathbf{x}^j, \mathbf{y}^j\}_{j=1}^B$

  // Adaptation loop ($\mathcal{O}$ in Figure 1c)

  for $j \leftarrow 1$ **to** B **do**

    // For 1 adaptation step

    $\boldsymbol{\phi}^j \leftarrow \boldsymbol{\phi}_0 - \alpha \nabla_{\boldsymbol{\phi}_0} \mathcal{L}_{\texttt{MSE}}(f(\mathbf{x}^j, \boldsymbol{\theta}, \boldsymbol{\phi}_0), \mathbf{y}^j)$

  // Update using adapted latent modulation

  $\boldsymbol{\phi}_0 \leftarrow \boldsymbol{\phi}_0 - \beta \mathbb{E}[\nabla_{\boldsymbol{\phi}_0} \mathcal{L}_{\texttt{MSE}}(f(\mathbf{x}^j, \boldsymbol{\theta}, \boldsymbol{\phi}^j), \mathbf{y}^j)]$

  // Remaining INR parameters

  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \beta \mathbb{E}[\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\texttt{MSE}}(f(\mathbf{x}^j, \boldsymbol{\theta}, \boldsymbol{\phi}^j), \mathbf{y}^j)]$

Result: Dataset of latent modulations $\{\boldsymbol{\phi}^i\}_{i=1}^N$, $\boldsymbol{\theta}$

Algorithm 2 Quantisation training stage

Data: Dataset of latent modulations $\{\boldsymbol{\phi}^i\}_{i=1}^N$, $\boldsymbol{\theta}$, λ

while *not converged* **do**

  Initialise parameters $\boldsymbol{\pi}_a, \boldsymbol{\pi}_s$.

  Sample batch of data $\mathcal{B} = \{\boldsymbol{\phi}^j, \mathbf{x}^j, \mathbf{y}^j\}_{j=1}^B$

  for $j \leftarrow 1$ **to** B **do**

    $\mathbf{z} \leftarrow g_a(\boldsymbol{\phi}^j; \boldsymbol{\pi}_a)$

    // Rounding at inference to obtain $\hat{\mathbf{z}}^j$

    $\widetilde{\mathbf{z}}^j = \mathbf{z}^j + \boldsymbol{\epsilon}; \boldsymbol{\epsilon} \sim \mathcal{U}(-\frac{1}{2}, \frac{1}{2})$

    // Compute entropy model $p_{\hat{\mathbf{z}}}$ and rate

    $\ell^j_{\texttt{rate}} = -\log_2[p_{\hat{\mathbf{z}}}(\widetilde{\mathbf{z}})]$

    $\widetilde{\boldsymbol{\phi}}^j \leftarrow g_s(\widetilde{\mathbf{z}}^j; \boldsymbol{\pi}_s)$

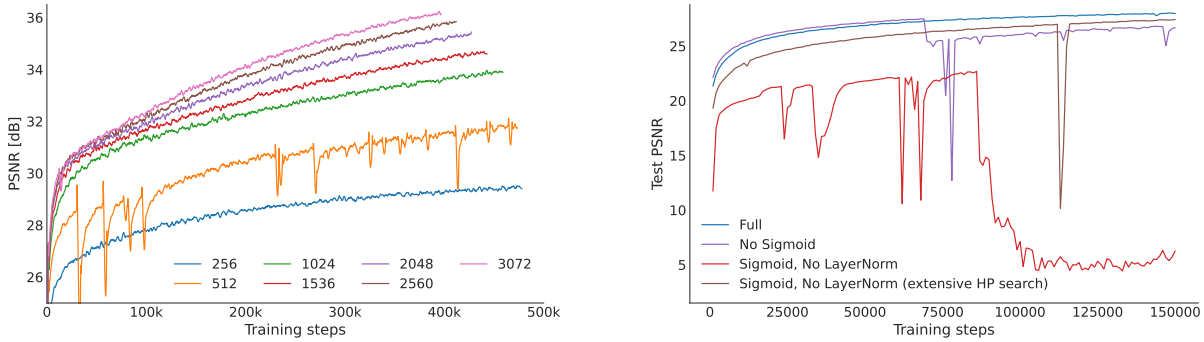
    $\ell^j_{\texttt{distortion}} = \mathcal{L}_{\texttt{MSE}}(f(\mathbf{x}^j, \boldsymbol{\theta}, \widetilde{\boldsymbol{\phi}}^j), \mathbf{y}^j)$

  $\boldsymbol{\pi}_a \leftarrow \boldsymbol{\pi}_a - \beta \mathbb{E}[\nabla_{\boldsymbol{\pi}_a}(\ell^j_{\texttt{rate}} + \lambda \ell^j_{\texttt{distortion}})]$

  $\boldsymbol{\pi}_s \leftarrow \boldsymbol{\pi}_s - \beta \mathbb{E}[\nabla_{\boldsymbol{\pi}_s}(\ell^j_{\texttt{rate}} + \lambda \ell^j_{\texttt{distortion}})]$

D. VC-INR algorithmic details

Algorithms 1 and 2 show details of the Meta-Learning (introduced in the previous section) and quantisation learning stages. The output of Algorithm 1 directly feeds into the pipeline for quantisation. Hence, the two problems of optimal parameterisation and quantisation can be tackled independently, thus allowing for various combination for future work.



(a) Effect of the projection layer width size

(b) Effect of LayerNorm

Figure 10. Performance during the meta-training phase. (a) investigates the effect of the width of the VC-INR non-linear projection layer and (b) compares the effect of LayerNorm on VC-INR.

E. Additional experimental results

E.1. Stabilising Meta-Learning of Soft Gating mask Modulations with LayerNorm

In Section 3.3, we demonstrate the importance of using LayerNorm (Ba et al., 2016) in the meta-learning of our new parameterisation. In Figure 10b we demonstrate that Meta-Learning becomes highly unstable by default (an effect becoming more severe with larger $\dim(\phi)$) and thus requires extensive hyperparameter search which may still suffer from occasional instability. Instead, we find that LayerNorm largely removes this phenomenon, leading to more stable training and better results. We hypothesise that such a divergence occurs when the norm of the inner loop gradient is large, indicating a sharp loss landscape. LayerNorm addresses this issue by smoothing the loss landscape, as has previously been shown (Santurkar et al., 2018; Xu et al., 2019). Furthermore, it effectively bounds the norm of ϕ.

In addition, we show that our new parameterisation can effectively make use of increasing network capacity (while Functa shows decreasing performance for non-linear mappings from latent parameters to modulations). Figure 10a shows this effect to be particularly effective for increasing network width, which we recommend for optimal performance during pre-training.

E.2. Results using MS-SSIM

In addition to results measured using PSNR, we provide results on Kodak using Multiscale structural similarity index measure (SSIM) (Wang et al., 2003) results in Figure 11 due to its better correlation with perceptual similarity. We observe comparable the results in Figure 4 with VC-INR performing similarly to JPEG & JPEG-2000.

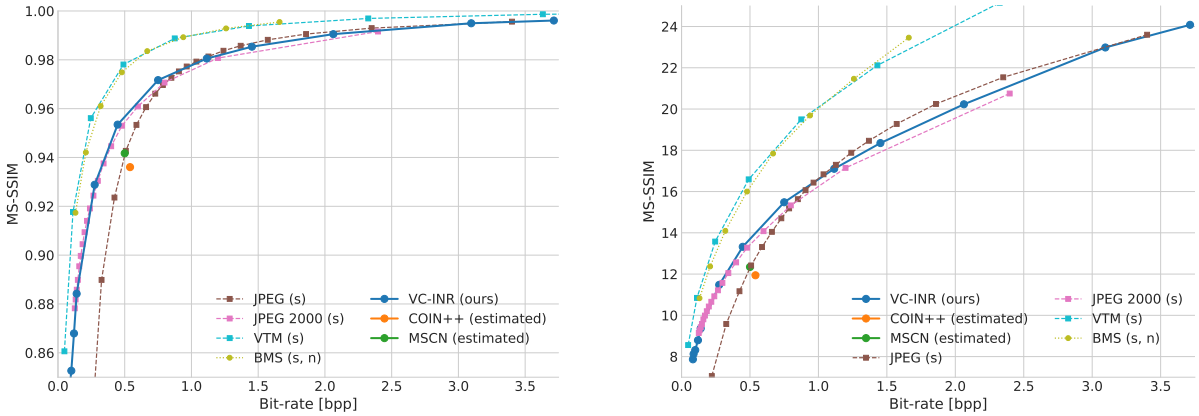


Figure 11. Compression results on image dataset Kodak measured using Multi-Scale Structural Similarity (MS-SSIM). (a) MS-SSIM scores, (b) Converted to decibel (i.e. $-10\log_{10}(1 - \text{MS-SSIM})$).

F. Qualitative Results

F.1. Cifar10

Figure 12 shows more qualitative results on Cifar10 for various rate/distortions trade-offs.

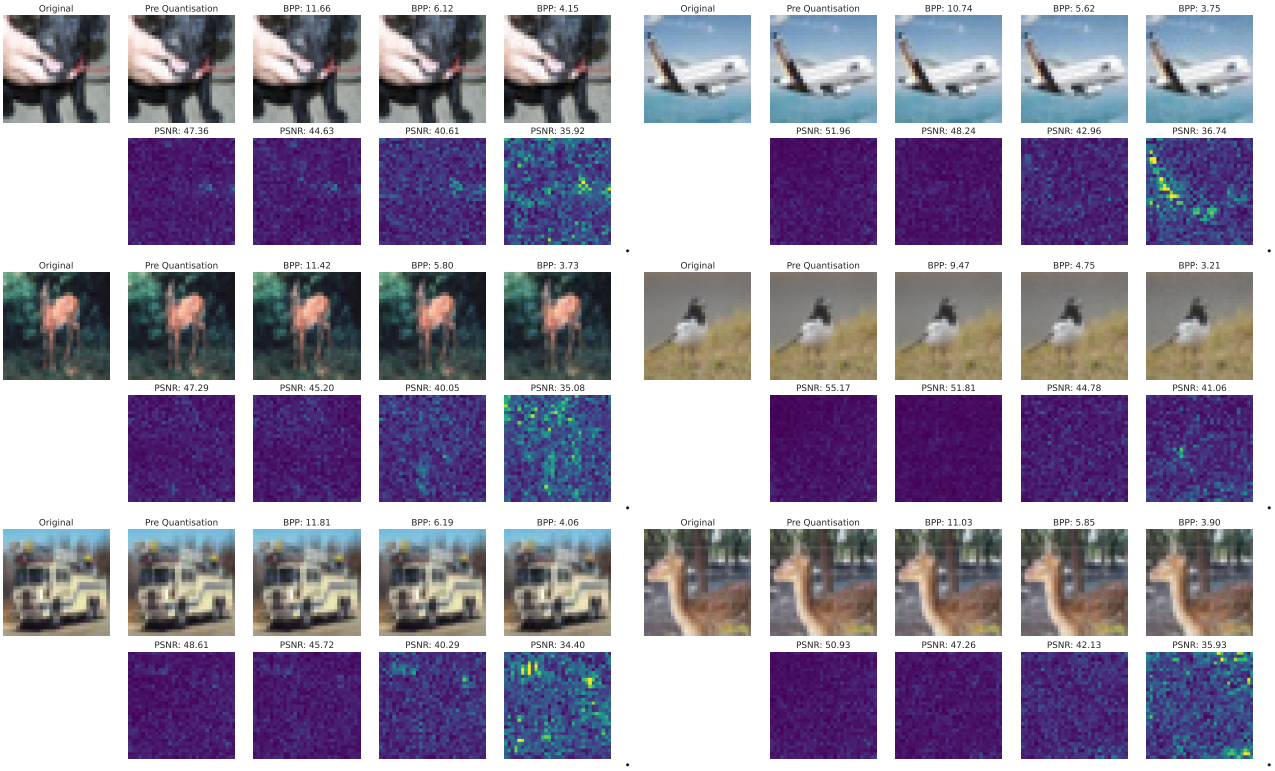


Figure 12. More qualitative results from the Cifar10 dataset.

In addition, we provide a further analysis of gating masks using a similar t-SNE projection as shown in the main text for Cifar-10 as well as an analysis of sparsity level and reconstruction correlation in Figure 13. With regards to correlation, it is firstly worth noting that there is little variation in the total sparsity level (reaching from 32.5 - 33.2). Secondly, we observe only very weak correlation (Pearson's correlation coefficient: 0.177) suggesting that no straightforward relationship between sparsity and performance exists.

F.2. Kodak

Figure 14 shows more qualitative results on Kodak in comparison with COIN++ (Dupont et al., 2022b) and MSCN (Schwarz & Teh, 2022).

F.3. UCF-101

Figure 15 shows more qualitative results on frames from the UCF-101 dataset. We provide links to each of the reconstruction video clips and its residual in comparison with the original video in Table 2.

G. Hyperparameters

G.1. Compression Experiments

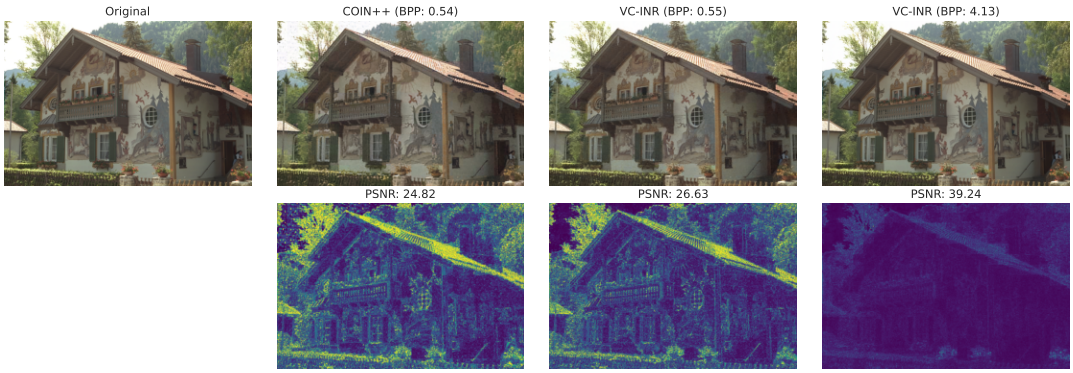
We show hyperparameters for both INR training and subsequent compression training for CIFAR-10 in Table 3, for Kodak in Table 4, for ERA5 in Table 5, for LibriSpeech in Table 6 and for UCF-101 in Table 7.

(a) (b) (c)

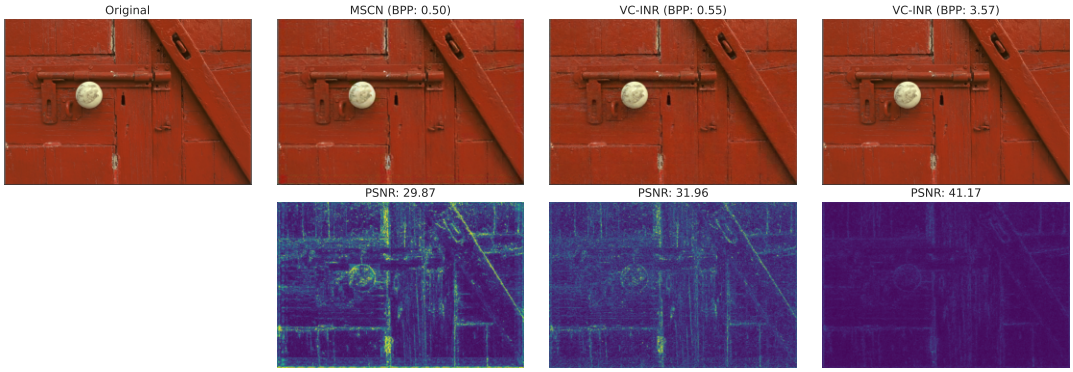
Figure 13. t-SNE (Van der Maaten & Hinton, 2008) visualisation of gating masks $\mathbf{G}_{\texttt{low}}$ after adaptation on CIFAR-10. (a) Full test set results (b) zoomed-in (c) Correlation between gating mask sparsity and performance level. Sparsity calculated as the fraction of sparse weights (i.e. $|(\mathbf{G}_{\texttt{low}} \odot \mathbf{W})_{ij}| < 0.001$) relative to the total number of all weights in the network.

	Quality			
	Low (BPP: 0.13)	Medium (BPP: 1.39)	High (BPP: 2.76)	Best (BPP: 4.20)
Example 1	here	here	here	here
Example 2	here	here	here	here
Example 3	here	here	here	here
Example 4	here	here	here	here
Example 5	here	here	here	here
Example 6	here	here	here	here
Example 7	here	here	here	here
Example 8	here	here	here	here
Example 9	here	here	here	here
Example 10	here	here	here	here
Example 11	here	here	here	here
Example 12	here	here	here	here
Example 13	here	here	here	here
Example 14	here	here	here	here
Example 15	here	here	here	here
Example 16	here	here	here	here

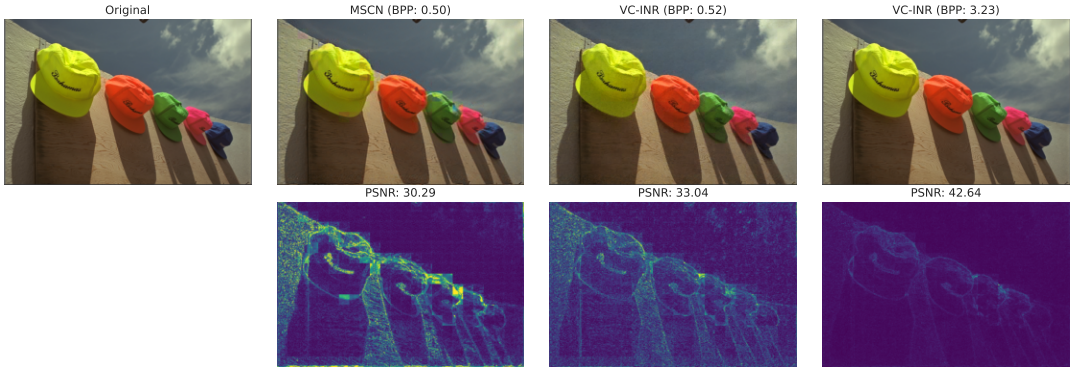
Table 2. More qualitative examples from the UCF-101 datasets. Shown are full video reconstructions and residuals of various VC-INR models at varying bit-rates.



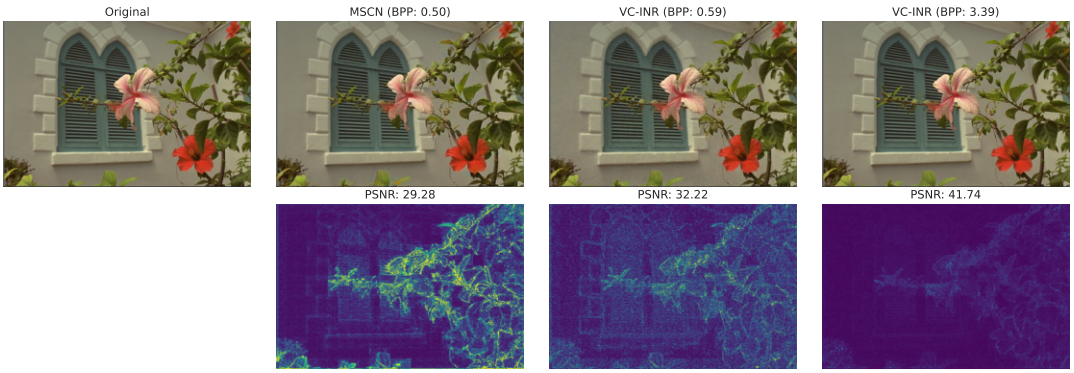
(a) Compared with COIN++ (Dupont et al., 2022b).



(b) Compared with MSCN (Schwarz & Teh, 2022).

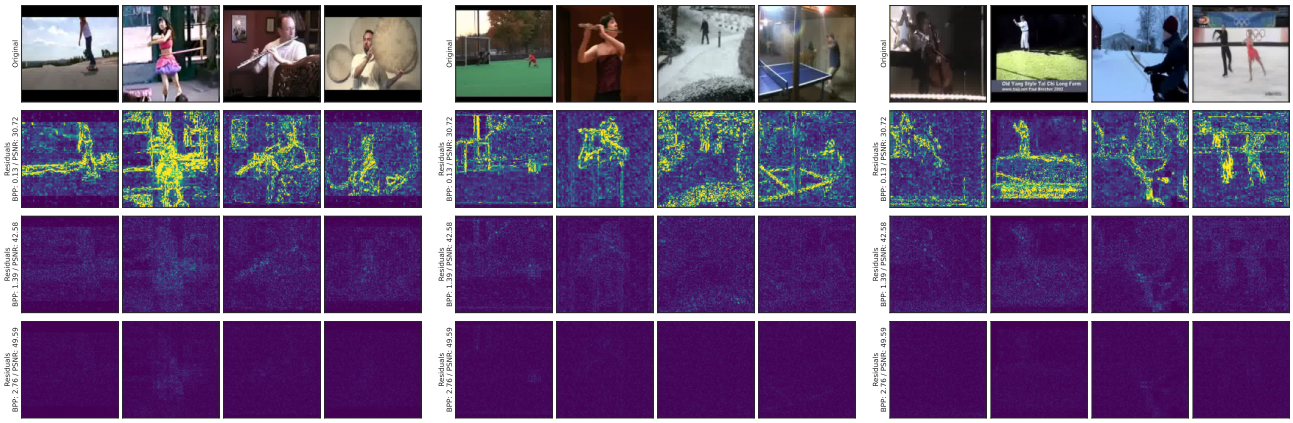


(c) Compared with MSCN (Schwarz & Teh, 2022).



(d) Compared with MSCN (Schwarz & Teh, 2022).

Figure 14. More qualitative results from the Kodak dataset. Shown are VC-INR models in comparison with other INR-based techniques at similar bit-rates (3rd column) as well as a high-quality model (last column).



(a) (b) (c)

Figure 15. More qualitative results from the UCF-101 dataset. Shown are VC-INR models at varying quality rates.

Table 3. Hyperparameters for compression experiments on CIFAR-10.

Parameter	**Considered range**	**Comment**
INR training		
Patching	{False}	
Activation function	$\{h(x) : \sin(\omega_0 x)$ (SIREN)$\}$	
ω_0	$\{30\}$	
Network depth	$\{15\}$	
Network width	$\{512\}$	
Batch size per device	$\{32, 64\}$	
Num devices	$\{8\}$	
Optimiser	{Adam}	
Outer learning rate	$\{3 \cdot 10^{-6}\}$	
Num inner steps	$\{3\}$	
Meta-learn $\boldsymbol{\phi}$ init.	{True}	
Meta SGD range	{[-5.0, 5.0]}	(Max./Min. for Meta-SGD LRs)
Meta SGD init range	{[1.0, 1.0]}	(Uniformly sampled).
$\boldsymbol{\phi} \rightarrow \{\mathbf{G}^{(1)}_{\texttt{low}}, \ldots, \mathbf{G}^{(L)}_{\texttt{low}}\}$ network		
$\dim(\boldsymbol{\phi})$	{2048, 3072, 4096}	
Use LayerNorm	{True}	
Network width	$\{6144\}$	
Residual blocks	$\{2\}$	
Activation function	{Leaky Relu}	(Xu et al., 2015)
Adapt first Layer	{False}	Apply low-rank gating to 1st layer?
Quantiser training		
Normalise $\boldsymbol{\phi}$	{True}	Per dim. $\frac{\phi_i - \hat{\mu}_i}{\hat{\sigma}_i}$ based on $\boldsymbol{\phi}$ train-set stats.
λ ($\mathcal{L}_{\texttt{distortion}}$ penalty)	{0.33, 0.66, 1.0, 3.33, 6.66}	
Analysis transform (g_a) Residual blocks	$\{1\}$	
g_a Network width	$\{2048, 4096, 5120\}$	
g_a Activation function	SeLU	(Klambauer et al., 2017)
$\dim(\mathbf{y})$	{1024, 2048, 4096, 5120}	
Synthesis transform g_s	Same as g_a	
Optimiser	{Adam}	
Learning rate	$\{1 \cdot 10^{-4}\}$	
Batch size per device	$\{32, 64\}$	
Num devices	$\{1\}$	

Table 4. Hyperparameters for compression experiments on Div2k/Kodak.

Parameter	Considered range	Comment
INR training		
Pre-training on	{Div2k}	as in (Schwarz & Teh, 2022; Strümpler et al., 2022)
Patching	$\{(32 \times 32)\}$	Dividing 768×512 images.
Activation function	$\{h(x) : \sin(\omega_0 x)$ (SIREN)$\}$	
ω_0	$\{30\}$	
Network depth	$\{15\}$	
Network width	$\{512\}$	
Batch size per device	$\{32\}$	
Num devices	$\{8\}$	
Optimiser	{Adam}	
Outer learning rate	$\{3 \cdot 10^{-6}\}$	
Num inner steps	$\{3\}$	
Meta-learn $\boldsymbol{\phi}$ init.	{True}	
Meta SGD range	{[-5.0, 5.0]}	(Max./Min. for Meta-SGD LRs)
Meta SGD init range	{[1.0, 1.0]}	(Uniformly sampled).
$\boldsymbol{\phi} \rightarrow \{\mathbf{G}^{(1)}_{\texttt{low}}, \ldots, \mathbf{G}^{(L)}_{\texttt{low}}\}$ **network**		
$\dim(\boldsymbol{\phi})$	{512, 1024}	
Use LayerNorm	{True}	
Network width	$\{4096\}$	
Residual blocks	$\{1\}$	
Activation function	{Leaky Relu}	(Xu et al., 2015)
Adapt first Layer	{False}	Apply low-rank gating to 1st layer?
Quantiser training		
Normalise $\boldsymbol{\phi}$	{True}	Per dim. $\frac{\boldsymbol{\phi}_i - \hat{\boldsymbol{\mu}}_i}{\hat{\boldsymbol{\sigma}}_i}$ based on $\boldsymbol{\phi}$ train-set stats.
λ ($\mathcal{L}_{\texttt{distortion}}$ penalty)	{0.01, 0.033, 0.1, 0.33, 0.66, 1.0}	
Analysis transform (g_a) Residual blocks	$\{1\}$	
g_a Network width	$\{256, 512, 1024\}$	
g_a Activation function	SeLU	(Klambauer et al., 2017)
$\dim(\mathbf{y})$	$\{256, 512, 1024\}$	
Synthesis transform g_s	Same as g_a	
Optimiser	{Adam}	
Learning rate	$\{1 \cdot 10^{-4}\}$	
Batch size per device	$\{128\}$	
Num devices	$\{1\}$	

Table 5. Hyperparameters for compression experiments on ERA5 (16×).

Parameter	Considered range	Comment
INR training		
Patching	{False}	
Activation function	$\{h(x) : \sin(\omega_0 x)$ (SIREN)$\}$	
ω_0	$\{30\}$	
Network depth	$\{10\}$	
Network width	$\{384\}$	
Batch size per device	$\{4\}$	
Num devices	$\{4\}$	
Optimiser	{Adam}	
Outer learning rate	$\{3 \cdot 10^{-6}\}$	
Num inner steps	$\{3\}$	
Meta-learn $\boldsymbol{\phi}$ init.	{True}	
Meta SGD range	{[-5.0, 5.0]}	(Max./Min. for Meta-SGD LRs)
Meta SGD init range	{[1.0, 1.0]}	(Uniformly sampled).
$\boldsymbol{\phi} \rightarrow \{\mathbf{G}^{(1)}_{\texttt{low}}, \ldots, \mathbf{G}^{(L)}_{\texttt{low}}\}$ network		
$\dim(\boldsymbol{\phi})$	{4, 8, 12, 32, 64, 128}	
Use LayerNorm	{True}	
Network width	$\{512\}$	
Residual blocks	$\{2\}$	
Activation function	{Leaky Relu}	(Xu et al., 2015)
Adapt first Layer	{False}	Apply low-rank gating to 1st layer?
Quantiser training		
Normalise $\boldsymbol{\phi}$	{True}	Per dim. $\frac{\phi_i - \hat{\mu}_i}{\hat{\sigma}_i}$ based on $\boldsymbol{\phi}$ train-set stats.
λ ($\mathcal{L}_{\texttt{distortion}}$ penalty)	{0.001, 0.01, 0.01, 0.1}	
Analysis transform (g_a) Residual blocks	$\{2\}$	
g_a Network width	$\{8, 12, 32, 64, 128\}$	
g_a Activation function	SeLU	(Klambauer et al., 2017)
$\dim(\mathbf{y})$	{8, 12, 32, 64, 128}	
Synthesis transform g_s	Same as g_a	
Optimiser	{Adam}	
Learning rate	$\{1 \cdot 10^{-4}\}$	
Batch size per device	$\{128, 256\}$	
Num devices	$\{1\}$	

Table 6. Hyperparameters for compression experiments on LibriSpeech.

Parameter	Considered range	Comment
INR training		
Patching	$\{(200, 400, 800)\}$	Dividing 48k dim. audio signal.
Activation function	$\{h(x) : \sin(\omega_0 x)$ (SIREN)$\}$	
ω_0	$\{10, 30, 50\}$	
Network depth	$\{10\}$	
Network width	$\{512\}$	
Batch size per device	$\{32, 64\}$	
Num devices	$\{1\}$	
Optimiser	{Adam}	
Outer learning rate	$\{3 \cdot 10^{-6}\}$	
Num inner steps	$\{3\}$	
Meta-learn $\boldsymbol{\phi}$ init.	{True}	
Meta SGD range	{[-5.0, 5.0]}	(Max./Min. for Meta-SGD LRs)
Meta SGD init range	{[1.0, 1.0]}	(Uniformly sampled).
$\boldsymbol{\phi} \rightarrow \{\mathbf{G}^{(1)}_{\texttt{low}}, \ldots, \mathbf{G}^{(L)}_{\texttt{low}}\}$ **network**		
$\dim(\boldsymbol{\phi})$	{64, 128, 256, 512, 1024}	
Use LayerNorm	{True}	
Network width	{512, 512, 768, 1536, 3072}	
Residual blocks	$\{2\}$	
Activation function	{Leaky Relu}	(Xu et al., 2015)
Adapt first Layer	{False}	Apply low-rank gating to 1st layer?
Quantiser training		
Normalise $\boldsymbol{\phi}$	{True}	Per dim. $\frac{\phi_i - \hat{\mu}_i}{\hat{\sigma}_i}$ based on $\boldsymbol{\phi}$ train-set stats.
λ ($\mathcal{L}_{\texttt{distortion}}$ penalty)	{1.0, 10.0, 100.0}	
Analysis transform (g_a) Residual blocks	$\{2\}$	
g_a Network width	$\{128, 256, 512, 1024\}$	
g_a Activation function	SeLU	(Klambauer et al., 2017)
$\dim(\mathbf{y})$	{128, 256, 512, 1024}	
Synthesis transform g_s	Same as g_a	
Optimiser	{Adam}	
Learning rate	$\{1 \cdot 10^{-4}\}$	
Batch size per device	$\{128\}$	
Num devices	$\{1\}$	

Table 7. Hyperparameters for compression experiments on UCF-101.

Parameter	Considered range	Comment
INR training		
Patching	$\{(4, 8, 8), (8, 8, 8), (4, 16, 16), (8, 16, 16)\}$	Dividing $(24, 128, 128)$ dim. video.
Activation function	$\{h(x) : \sin(\omega_0 x)$ (SIREN)$\}$	
ω_0	$\{30\}$	
Network depth	$\{10\}$	
Network width	$\{256\}$	
Batch size per device	$\{4\}$	
Num devices	$\{4\}$	
Optimiser	{Adam}	
Outer learning rate	$\{3 \cdot 10^{-6}\}$	
Num inner steps	$\{3\}$	
Meta-learn $\boldsymbol{\phi}$ init.	{True}	
Meta SGD range	{[-5.0, 5.0]}	(Max./Min. for Meta-SGD LRs)
Meta SGD init range	{[1.0, 1.0]}	(Uniformly sampled).
$\boldsymbol{\phi} \rightarrow \{\mathbf{G}_{\texttt{low}}^{(1)}, \ldots, \mathbf{G}_{\texttt{low}}^{(L)}\}$ **network**		
$\dim(\boldsymbol{\phi})$	{512, 1536, 2048, 2048}	
Use LayerNorm	{True}	
Network width	$\{512\}$	
Residual blocks	$\{2\}$	
Activation function	{Leaky Relu}	(Xu et al., 2015)
Adapt first Layer	{False}	Apply low-rank gating to 1st layer?
Quantiser training		
Normalise $\boldsymbol{\phi}$	{True}	Per dim. $\frac{\boldsymbol{\phi}_i - \hat{\boldsymbol{\mu}}_i}{\hat{\boldsymbol{\sigma}}_i}$ based on $\boldsymbol{\phi}$ train-set stats.
λ ($\mathcal{L}_{\texttt{distortion}}$ penalty)	{0.001, 0.01, 0.1, 1.0, 10.0}	
Analysis transform (g_a) Residual blocks	$\{1\}$	
g_a Network width	$\{256, 512, 1024, 2048\}$	
g_a Activation function	SeLU	(Klambauer et al., 2017)
$\dim(\mathbf{y})$	{256, 512, 1024, 2048}	
Synthesis transform g_s	Same as g_a	
Optimiser	{Adam}	
Learning rate	$\{1 \cdot 10^{-4}\}$	
Batch size per device	$\{64\}$	
Num devices	$\{1\}$	