Efficient Meta-Learning via Error-based Context Pruning for Implicit Neural Representations

Jihoon Tack [1] **Subin Kim** [1] **Sihyun Yu** [1] **Jaeho Lee** [2] **Jinwoo Shin** [1] **Jonathan Richard Schwarz** [3]

Abstract

We introduce an efficient optimization-based meta-learning technique for learning large-scale implicit neural representations (INRs). Our main idea is designing an online selection of context points, which can significantly reduce memory requirements for meta-learning in any established setting. By doing so, we expect additional memory savings which allows longer per-signal adaptation horizons (at a given memory budget), leading to better meta-initializations by reducing myopia and, more crucially, enabling learning on high-dimensional signals. To implement such context pruning, our technical novelty is threefold. First, we propose a selection scheme that adaptively chooses a subset at each adaptation step based on the predictive error, leading to the modeling of the global structure of the signal in early steps and enabling the later steps to capture its high-frequency details. Second, we counteract any possible information loss from context pruning by minimizing the parameter distance to a bootstrapped target model trained on a full context set. Finally, we suggest using the full context set with a gradient scaling scheme at test-time. Our technique is model-agnostic, intuitive, and straightforward to implement, showing significant reconstruction improvements for a wide range of signals. Code is available at https://github.com/jihoontack/ECoP

1. Introduction

Implicit neural representations (INRs) have emerged as a new paradigm for representing complex signals as continuous coordinate-mapping functions (e.g., image as a $(x, y) \mapsto (r, g, b)$ mapping) parameterized by neural networks. This approach has shown great promise in various modalities, including images (Martel et al., 2021; Müller et al., 2022), videos (Chen et al., 2021), 3D scenes (Park et al., 2019), and audio (Dupont et al., 2022b), due to their numerous intriguing properties accompanied by the parameter efficiency. However, fitting an INR for a single signal is even too costly (e.g., more than a half GPU day for a single HD video; Kim et al., 2022), limiting the scalability of learning INRs for a large set of signals. To address this limitation, optimization-based meta-learning has gained their attention to accelerate learning (Sitzmann et al., 2020a; Tancik et al., 2021; Lee et al., 2021), rapidly improving the reconstruction of signals in a few optimization steps and thus becoming a standard technique for INR research.

Unfortunately, such optimization-based meta-learning techniques scale very poorly with the dimensionality of the signal, as coordinate and signal value pairs, often called as context set, for the optimization increase super-linearly with respect to the dimension.[1] This is problematic since firstly, a suitable number of gradient steps typically increases with the size of the context set, and secondly, prior schemes require the entire context set to be fitted at once. Meta-learning at scale thus quickly becomes prohibitively expensive in memory requirements. To this end, recent attempts propose to divide the signal into a set of low-resolution patches (Dupont et al., 2022b; Schwarz & Teh, 2022). However, such a method notably increases adaptation time (i.e., proportional to the number of patches) and ignores the cross-patch statistics, which often leads to inefficiency, e.g., modeling redundant patches in the signal.

Instead, drawing inspiration from the recent advances in data pruning literature (Paul et al., 2021), we ask: *Can we effectively reduce the size of the context set for the optimization without compromising the adaptation performance?*

Contribution. We propose Error-based Context Pruning (ECoP), an efficient and effective optimization-based meta-learning framework for scalable INR learning. Specifically, ECoP involves online sub-sampling of the context set based on the error (or the loss) of each element, making memory efficiency first-class citizens in algorithm design. As ECoP focuses on high-loss elements at each step of adapta-

[1]KAIST, Daejeon, South Korea [2]POSTECH, Pohang, South Korea [3]University College London, London, United Kingdom. Correspondence to: Jihoon Tack <jihoontack@kaist.ac.kr>.

[1]A single 1024×1024 high-resolution image is thus interpreted as a context set over a million $((x, y), (r, g, b))$ pairs.

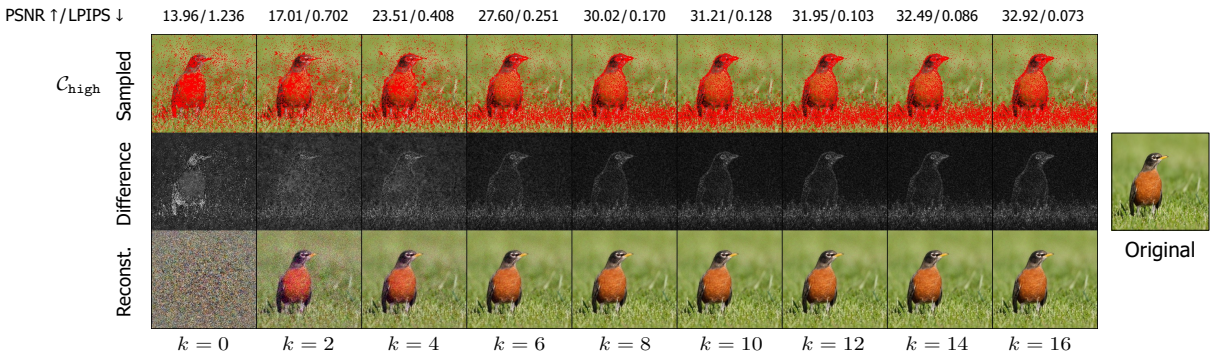


Figure 1. Visualization of sampled points (first), the difference between the original signal (middle), and the reconstructed signal (last) via ECoP trained on ImageNet-100 with SIREN. The sampled coordinates are highlighted in red where the sampling ratio γ is 0.25, and k denotes the adaptation step. ECoP first focuses on the global structure and then models the high-frequency features of the signal.

tion, it effectively sub-samples based on the feature statistics, first focusing on the global structure before modeling high-frequency details of the signal (see Figure 1). Furthermore, these savings in memory requirements enables longer adaptation horizons to learn a better meta-initialization (given a memory budget) and more crucially, make it possible to train on higher-dimensional signals.

To implement such context pruning during meta-learning INRs, we argue two considerations should be taken into account: (a) minimizing the possible information loss due to the pruned context and (b) using the full context set (rather than the pruned set) for the meta-testing. To this end, we propose two additional techniques.

- *Bootstrapped correction*: To correct any possible information loss introduced by the context pruning, we propose to nudge the parameters learned using pruned context sets to be as close as possible to the result obtained using the full set. Specifically, we generate the bootstrapped target model by continuing the adaptation from the meta-learner using the full context set, then minimize the parameter distance between the two models. Importantly, this correction introduces significantly less computational overhead than the meta-learning procedure, as it is unnecessary to save any intermediate gradients (i.e., second-order gradients) for generating the target.
- *Test-time gradient scaling*: We also found that naïve use of a full context set at meta-testing does not work as the corresponding gradient norm, i.e., the magnitude of gradient update, is different from that of pruned contexts used at meta-training. To address this, we propose scaling the test-time gradient using the ratio of gradient norms between the pruned and full context sets.

We verify the efficacy of ECoP through extensive evaluations on various data modalities, including image, video, audio, and manifold datasets. Overall, our experimental results demonstrate strong results, consistently and significantly outperforming previous meta-learning approaches in signal reconstruction. For instance, measured with peak signal-to-noise ratio (PSNR), ECoP improves the prior state-of-the-art results by 38.28 $\rightarrow$ 40.54 on CelebA (Liu et al., 2015), 28.86 $\rightarrow$ 33.99 on UCF-101 (Soomro et al., 2012), and 31.39 $\rightarrow$ 36.45 on Librispeeh (Panayotov et al., 2015), respectively. Furthermore, we demonstrate that ECoP could even meta-learn on high-dimensional signals (e.g., a video of 256×256×32) where prior works suffer from a memory shortage under the same machine.

2. ECoP: Meta-Learning via Error-based Context Pruning

In this section, we present Error-based Context Pruning (ECoP), an efficient meta-learning scheme for large-scale implicit neural representation (INR) learning. We first review meta-learning for INRs (Section 2.1), and then present the core components of ECoP: (i) efficient online context pruning (Section 2.2), error correction with a bootstrapped target (Section 2.3), and meta-testing with full context set (Section 2.4). We provide the overview and pseudocode of ECoP in Figure 2 and Algorithm 1, respectively.

2.1. Meta-Learning Implicit Neural Representations

Consider given N signals $\mathbf{s}_1, \dots, \mathbf{s}_N$, where each $\mathbf{s}$ is represented as a context set $\mathcal{C} := \{(\mathbf{x}_j, \mathbf{y}_j)\}_{j=1}^{M}$ consisting of M coordinate-value pairs $(\mathbf{x}_j, \mathbf{y}_j)$ with a coordinate $\mathbf{x}_j \in \mathbb{R}^C$ and a signal value $\mathbf{y}_j \in \mathbb{R}^D$.[2] We are interested in finding parameters $\theta^{(1)}, \dots, \theta^{(N)}$ of corresponding N INRs $f_{\theta^{(1)}}, \dots, f_{\theta^{(N)}}$ (respectively); where each INR $f_{\theta^{(i)}} : \mathbb{R}^C \rightarrow \mathbb{R}^D$ well approximates $\mathbf{s}_i$. A standard choice of learning these parameters is to optimize each θ_i to represent corresponding $\mathbf{s}_i$ independently with mean-squared error (MSE), i.e., $\mathcal{L}_{\texttt{MSE}}(\theta; \mathcal{C}) := \frac{1}{M}\sum_{j=1}^{M} ||f_\theta(\mathbf{x}_j) - \mathbf{y}_j||_2^2$. However, it requires either significant computations or memory requirements and thus limits the scalability if N is large or signals are high-dimensional (i.e., M is large).

[2] If suitable, we write $\mathcal{C}$ as $\mathcal{C}_{\texttt{full}}$ to note it is full context set.

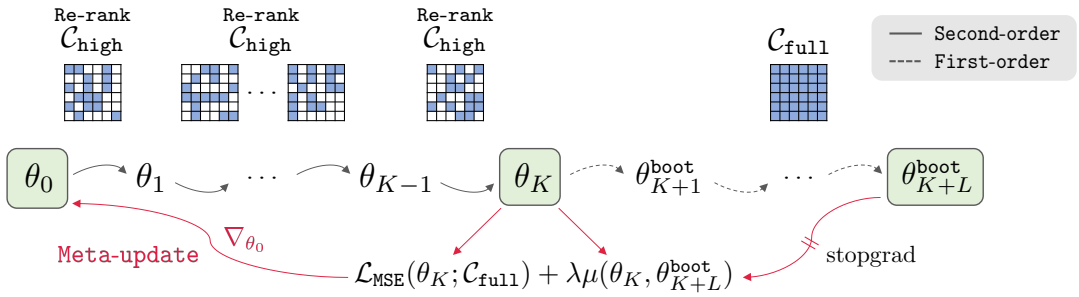


Figure 2. Computational diagram of ECoP. A meta-learned initialization θ_0 is adapted for K steps to obtain θ_K, re-ranking and pruning the context set $\mathcal{C}_{\texttt{high}}$ at each step for the memory efficiency. Subsequently, we create a target bootstrap model $\theta^{\texttt{boot}}_{K+L}$ by updating for L additional steps, now using *the full context set* $\mathcal{C}_{\texttt{full}}$. A meta-loss is computed according to a distance metric $\mu(\theta_K, \theta^{\texttt{boot}}_{K+L})$ between the two parameters and the reconstruction error of the meta-learner $\mathcal{L}_{\texttt{MSE}}(\theta_K;\mathcal{C}_{\texttt{full}})$. θ_0 is then updated to allow the minimization of this distance in the original K steps, correcting for the pruning of the context set and leading to a better overall initialization.

Remarkably, this process is frequently accelerated by taking meta-learning approaches: training a shared initialization θ_0 from which each signal $\mathbf{s}$ and its corresponding context set $\mathcal{C}$ can be well approximated within a few gradient steps (Tancik et al., 2021). This is primarily done with model-agnostic meta-learning (MAML; Finn et al., 2017):

$$\theta_0 = \min_{\theta_0} \mathbb{E}_{\mathcal{C}\sim\hat{p}(\mathcal{C})}[\mathcal{L}_{\texttt{MSE}}(\theta_0 - \alpha\nabla_{\theta_0}\mathcal{L}_{\texttt{MSE}}(\theta_0;\mathcal{C});\mathcal{C})] \quad (1)$$

where both outer and inner optimization problems are solved through iterative gradient descent. The inner optimization is typically iterated for K steps before a meta-optimization (or outer) step w.r.t θ_0 is taken, where $\hat{p}(\mathcal{C})$ is the (empirical) distribution over context sets, each representing a signal.

However, this approach faces severe memory inefficiency and thus limits the scalability to high-resolution signals. This is because the above Eq. (1) requires a memory proportional to the context set size M, where M increases strictly with the signal resolution (possibly over a million).[3] Furthermore, prior works (Tancik et al., 2021; Dupont et al., 2022a) have exhibited a necessity of second-order optimization in Eq. (1) for meta-learning INRs, which requires the respective activations and computation graph to be kept in the memory; it prevents using memory efficient first-order meta-learning methods (Finn et al., 2017; Nichol et al., 2018) and exacerbates the memory inefficiency. Such a second-order optimization also forces the method to use small K due to increased memory usage, which leads to overall performance degradation by falling into myopia.

2.2. Error-based Online Context Pruning

To alleviate memory limitations in meta-learning INRs, we suggest an *online* context-pruning strategy during the training stage which significantly reduces the memory usage but does not degrade the performance too much: at each inner loop iteration, our strategy *adaptively* chooses high-error coordinate-value pairs (for the adaptation) to reduce the burden in memory intensive inner optimization.

Our design choice is inspired by data pruning, a recent surge in interest in the field of machine learning, which sub-samples a large training dataset to reduce unnecessary storage and training costs. Here, a recent work suggests the data-wise error of early training stage neural networks can be a powerful criterion for data pruning (EL2N score; Paul et al., 2021). Concretely, given a training iteration k and a data point $(\mathbf{x}, \mathbf{y})$, EL2N score $R_k(\cdot, \cdot)$ is estimated with the expectation of the error over the neural network parameters θ_k at iteration k, where such expectation can be empirically estimated with a single model (Sorscher et al., 2022) as:

$$R_k(\mathbf{x}, \mathbf{y}) \coloneqq \|f_{\theta_k}(\mathbf{x}) - \mathbf{y}\|_2. \quad (2)$$

Then, one can prune the data with small R_k values in one-shot from the dataset. Due to its promise, utilizing the EL2N score is an attractive option for context pruning; however, exploiting it directly in our problem setup poses a scalability issue. Specifically, one should train models per-signal individually with a large enough iteration to adapt EL2N, resulting in severe computation burdens especially dealing with a large set of context sets as in our setup (i.e., large N).

We circumvent this issue by putting the error-based context pruning principle *inside* the meta-training stage: namely, we reduce a given context set at every inner step iteration $k \leq K$ for the next update, leveraging MAML's ability to rapidly absorb the information within a few gradient steps. Such a protocol also lets the pruning be done in *online* manner, i.e., re-ranking of scores and subsequent pruning of examples at each step from the full context set $\mathcal{C}_{\texttt{full}}$ according to their ranking, and thus focusing dynamically on pruning a better set for an update of θ_k.

Formally, for a given $\mathcal{C}_{\texttt{full}}$, our error-based context pruning

[3] Conventional few-shot learning setups in other domains consider relatively small context sets with $M \leq 50$.

Algorithm 1 Meta-training of ECoP

Input: $\{\mathbf{s}_i\}_{i=1}^N, \gamma, \alpha, \beta, \lambda, K, L$

Initialize θ_0.
while not converge **do**
  Sample batch $\{\mathbf{s}_1, \dots, \mathbf{s}_B\}$.
  for all $b = 1$ to B **do**
    Extract context $\mathcal{C}_{\texttt{full}}$ from $\mathbf{s}_b$.
    for all $k = 0$ to $K-1$ **do**
      `# Error-based context pruning`
      $\mathcal{C}_{\texttt{high}} = \text{Top}(\mathcal{C}_{\texttt{full}};\, R_k, \gamma)$
      `# Adaptation step`
      $\theta_{k+1} \leftarrow \theta_k - \alpha \nabla_{\theta_k} \mathcal{L}_{\texttt{MSE}}(\theta_k; \mathcal{C}_{\texttt{high}})$
    end for
    `# Generate target`
    $\theta^{\texttt{boot}}_{K+1} \leftarrow \theta_K - \alpha \nabla_{\theta_K} \mathcal{L}_{\texttt{MSE}}(\theta_K; \mathcal{C}_{\texttt{full}})$
    `...# Repeat L-1 times`
    $\theta^{\texttt{boot}}_{K+L} \leftarrow \text{stopgrad}(\theta_{K+L})$
    $\mathcal{L}^b_{\texttt{total}} = \mathcal{L}_{\texttt{MSE}}(\theta_K; \mathcal{C}_{\texttt{full}}) + \lambda \mu(\theta_K, \theta^{\texttt{boot}}_{K+L})$
  end for
  $\theta_0 \leftarrow \theta_0 - \beta \frac{1}{B} \sum_{b=1}^B \nabla_{\theta_0} \mathcal{L}^b_{\texttt{total}}$
end while

scheme is defined with a hyper-parameter $\gamma \in (0, 1)$:

$$\mathcal{C}^k_{\texttt{high}} := \text{Top}(\mathcal{C}_{\texttt{full}};\, R_k, \gamma) \tag{3}$$

where Top returns the elements with $\gamma|\mathcal{C}_{\texttt{full}}|$ highest R_k scores. The sampling ratio γ thus controls over the tradeoff between (expected) performance and memory requirements.

With the pruned context set $\mathcal{C}^k_{\texttt{high}}$ at iteration k, we adapt the meta-learner via gradient descent with a step size $\alpha > 0$:

$$\theta_{k+1} = \theta_k - \alpha \nabla_{\theta_k} \mathcal{L}_{\texttt{MSE}}(\theta_k; \mathcal{C}^k_{\texttt{high}}). \tag{4}$$

One of the intriguing aspects of our online context pruning procedure for INRs is that it automatically samples the global structure at first and then selects high-frequency details of the signal (see Figure 1), where such learned sampling resembles the hand-crafted technique for efficient INR training in previous literature (Landgraf et al., 2022).

2.3. Bootstrap Correction

Despite the careful selection of $\mathcal{C}^k_{\texttt{high}}$, reducing the context set $\mathcal{C}_{\texttt{full}}$ may introduce information loss. To tackle this issue, we suggest regularizing the parameter adapted with the prune context set to be as close to the parameter adapted with the *full context set* based on extending the idea of bootstrapped meta-learning (Flennerhag et al., 2022). Specifically, after adapting the INR meta-learner for K steps with the pruned context set (i.e., θ_K), we additionally adapt L step with the full context set to generate the bootstrapped target model $\theta^{\texttt{boot}}_{K+L}$ which is thus expected to show superior performance. The meta-learner then learns to minimize the distance between two models.

Formally, given a pre-defined distance function $\mu(\cdot\,, \cdot)$, we regularize θ_K to minimize the distance to the bootstrapped target's parameter $\theta^{\texttt{boot}}_{K+L}$ (Flennerhag et al., 2022), namely:

$$\mu(\theta_K, \theta^{\texttt{boot}}_{K+L}), \tag{5}$$

where we use ℓ_2 distance for μ. In practice, we use this regularization by propagating gradients to θ_K only by operating a stopgradient operation on $\theta^{\texttt{boot}}_{K+L}$.

Note that the bootstrapped correction can be achieved without consuming additional memory burden a lot, as we do not take second-order gradients w.r.t the bootstrapped parameters for the optimization of Eq. (1). Furthermore, bootstrapping introduces an additional benefit, as it extends the meta-learning horizon larger than K steps without requiring backpropagation through all $K + L$ updates, reducing myopia induced by the typically short K-step horizon.

Overall meta-learning objective. In practice, we find it is useful to calculate the final loss as a combination of the bootstrap correction term and the performance of θ_K on the full context set. For a given hyper-parameter $\lambda > 0$, the meta-objective of ECoP becomes:

$$\mathcal{L}_{\texttt{total}}(\theta_0; \mathcal{C}_{\texttt{full}}) := \mathcal{L}_{\texttt{MSE}}(\theta_K; \mathcal{C}_{\texttt{full}}) + \lambda \mu(\theta_K, \theta^{\texttt{boot}}_{K+L}).$$

2.4. Meta-test with Full Context Set

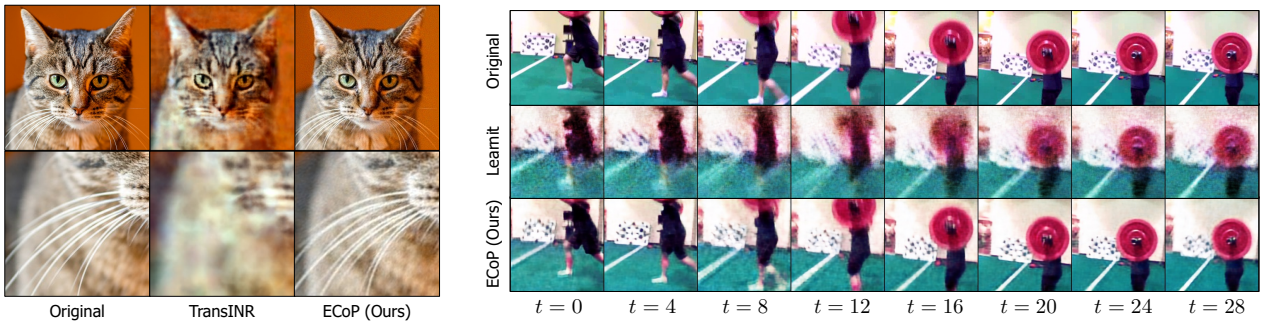
While we use the pruned context set for meta-training, one can use the full context set memory efficiently during the meta-test time by using the first-order adaptation. However, the problem is the norm of the gradients deviates a lot from meta-training and testing, i.e., $||\nabla_\theta \mathcal{L}_{\texttt{MSE}}(\theta; \mathcal{C}_{\texttt{high}})||_2 > ||\nabla_\theta \mathcal{L}_{\texttt{MSE}}(\theta; \mathcal{C}_{\texttt{full}})||_2$ by design of $\mathcal{C}_{\texttt{high}}$, resulting in a significant performance degradation. Accordingly, we suggest a simple remedy to scale the test-time gradient at step k:

$$g^{\texttt{test}}_k = \frac{||\nabla_{\theta_k} \mathcal{L}_{\texttt{MSE}}(\theta_k; \mathcal{C}_{\texttt{high}})||_2}{||\nabla_{\theta_k} \mathcal{L}_{\texttt{MSE}}(\theta_k; \mathcal{C}_{\texttt{full}})||_2} \nabla_{\theta_k} \mathcal{L}_{\texttt{MSE}}(\theta_k; \mathcal{C}_{\texttt{full}}), \tag{6}$$

so the parameter θ_k is updated with $g^{\texttt{test}}_k$ instead of $\nabla_{\theta_k} \mathcal{L}_{\texttt{MSE}}(\theta_k; \mathcal{C}_{\texttt{full}})$. We observe that such a simple refinement improves the test-time performance significantly.

3. Related Work

Implicit neural representations (INRs). INRs emerged as a new paradigm for representing complex, continuous signals (Sitzmann et al., 2020b; Mildenhall et al., 2020) as their number of parameters do not strictly scale with the resolution of the signal (Mescheder et al., 2019; Sitzmann et al., 2019), easing modeling of multi-modal signals (Du et al., 2021; Luo et al., 2022) and showing potential for new approaches to prominent applications and downstream tasks, including data compression (Dupont et al., 2021), classification (Dupont et al., 2022a), and generative modeling



(a) AFHQ (512×512) on SIREN (b) UCF-101 (256×256×32) on NeRV

Figure 3. Qualitative comparison between ECoP and baselines on high-resolution (a) AFHQ and (b) UCF-101 datasets.

(Skorokhodov et al., 2021; Yu et al., 2022). However, fitting INRs is quite costly (especially for high-resolution signals; Kim et al., 2022). In this paper, we develop an efficient meta-learning framework for large-scale INR training.

Efficient meta-learning. There have been several works in developing (memory) efficient algorithms in the field of meta-learning. Typically, such algorithms have been explored in amortization-based (or encoder-based) schemes, including Prototypical Networks (Snell et al., 2017; Bronskill et al., 2021), and Neural Processes (Garnelo et al., 2018b;a; Galashov et al., 2019). Unlike optimization-based meta-learning, however, these methods are somewhat limited in applicability to diverse modalities and INR architectures (requiring modality and model-specific design).

In an optimization-based regime, several memory efficient schemes were introduced, including first-order MAML (Finn et al., 2017), Reptile (Nichol et al., 2018), Implicit MAML (Rajeswaran et al., 2019) and continual trajectory shifting (Shin et al., 2021), where they do not use the second-order gradient adaptation. However, recent work has shown that first-order optimization-based schemes can underperform for INRs (Dupont et al., 2022a) where we also observed a similar result (in Table 5). Recently, sparsity has been combined when meta-learning INRs (Lee et al., 2021; Schwarz & Teh, 2022), which may help reduce computation at inference time but still requires some memory usage when meta-learning and may reduce the network expressive power. In this paper, we focus on developing a memory efficient meta-learning framework that is built upon second-order gradient-based schemes for effective INR learning.

Sparse data selection. ECoP is related to areas of machine learning, which focus on identifying subsets of a dataset. For instance, data pruning primarily focuses on designing a pruning metric to reduce the dataset without comprising the performance (Toneva et al., 2019; Feldman & Zhang, 2020), memory-based techniques of continual learning, where subsets of past tasks are used to prevent catastrophic forgetting (Titsias et al., 2020; Rolnick et al., 2019), and active learning which involves identifying and labeling data points to facilitate efficient learning progress during subsequent online updates (Sener & Savarese, 2018; Emam et al., 2021). In this paper, we develop an online context pruning scheme for meta-learning by drawing a connection with recent work of data pruning literature, i.e., EL2N (Paul et al., 2021).

4. Experiments

We extensively validate the effectiveness of the proposed ECoP by measuring its reconstruction performance on various dataset across modalities. We describe the experimental setups in Section 4.1, then present the main experimental results in Section 4.2. Finally, we conduct ablation studies and analysis of ECoP in Section 4.3.

4.1. Experimental Setups

In this section, we briefly provide the overall experimental setups. See Appendix A for further details of training, evaluation, architecture, data pre-processing, and resources.

Baselines. For the main experiments, we mainly compare ECoP with existing meta-learning schemes for INRs, including Learnit (Tancik et al., 2021) and TransINR (Chen & Wang, 2022), and also provide comparisons with efficient meta-learning methods such as FOMAML (Finn et al., 2017) and Reptile (Nichol et al., 2018). Additionally, we consider random initialization (that does not learn the initialization) as baselines to highlight the effectiveness of meta-learning. For TransINR, we use the official implementation and hyperparameters to reproduce the results, whereas, for Learnit, we increase the inner adaptation steps with ours for a fair comparison under the same memory budgets.

Evaluation setup. Unless otherwise specified, we use the same adaptation steps in test-time for both Learnit and ECoP. For quantitative evaluation, we mainly report peak signal-to-noise ratio (PSNR; higher is better) for the reconstruction performance and also measure the perceptual similarity for image and video datasets by using LPIPS (Zhang et al., 2018; lower is better) and SSIM (Wang et al., 2004; higher is better). For video datasets, we evaluate these metrics in

Table 1. Reconstruction performance of SIREN on image datasets of various resolutions. We report PSNR (dB), SSIM, and LPIPS, where the same adaptation steps were used for optimization-based meta-learning schemes and random initialization. N/A denotes the out-of-memory on a single NVIDIA A100 40GB GPU, and the bold indicates the best results of each group.

Resolution	Dataset	Method	PSNR (↑)	SSIM (↑)	LPIPS (↓)
178×178	CelebA	Random initialization	19.94	0.532	0.708
		Learnit (Tancik et al., 2021)	38.28	0.964	0.010
		TransINR (Chen & Wang, 2022)	32.37	0.913	0.068
		ECoP (Ours)	**40.54**	**0.975**	**0.005**
	Imagenette	Random initialization	18.57	0.443	0.810
		Learnit (Tancik et al., 2021)	35.66	0.950	0.014
		TransINR (Chen & Wang, 2022)	28.58	0.850	0.165
		ECoP (Ours)	**37.71**	**0.965**	**0.007**
	Text	Random initialization	15.73	0.574	0.755
		Learnit (Tancik et al., 2021)	30.31	0.956	0.018
		TransINR (Chen & Wang, 2022)	22.70	0.898	0.085
		ECoP (Ours)	**33.11**	**0.968**	**0.009**
256×256	ImageNet-100	Random initialization	18.30	0.439	0.855
		Learnit (Tancik et al., 2021)	30.92	0.868	0.130
		TransINR (Chen & Wang, 2022)	27.78	0.821	0.195
		ECoP (Ours)	**31.98**	**0.891**	**0.079**
512×512	CelebA-HQ	Random initialization	19.71	0.636	0.747
		Learnit (Tancik et al., 2021)	32.16	0.851	0.249
		TransINR (Chen & Wang, 2022)	28.27	0.798	0.299
		ECoP (Ours)	**33.42**	**0.874**	**0.211**
	AFHQ	Random initialization	18.57	0.488	0.856
		Learnit (Tancik et al., 2021)	28.58	0.751	0.354
		TransINR (Chen & Wang, 2022)	23.43	0.592	0.573
		ECoP (Ours)	**29.37**	**0.784**	**0.285**
1024×1024	CelebA-HQ	Random initialization	12.21	0.574	0.820
		Learnit (Tancik et al., 2021)	27.66	0.781	0.513
		TransINR (Chen & Wang, 2022)	N/A	N/A	N/A
		ECoP (Ours)	**28.89**	**0.789**	**0.439**

a frame-wise manner and average over the whole video by following the prior works (Chen et al., 2021).

Architectures. We use SIREN (Sitzmann et al., 2020b) as the base architecture for all experiments, and additionally consider NeRV (Chen et al., 2021) for the video domain.

Datasets. For the main experiment, we consider four different modalities, including image, video, audio, and manifold datasets. For image datasets, we follow the experimental setup of Learnit and use CelebA (Liu et al., 2015), Imagenette (Howard, 2019), and Text (Tancik et al., 2021) datasets. We additionally consider the high-resolution multi-class dataset, i.e., ImageNet-100 (Tian et al., 2020), and high-resolution fine-grained datasets, including CelebA-HQ (Karras et al., 2018) and AFHQ (Choi et al., 2020). For the video dataset, we use UCF-101 (Soomro et al., 2012) with two different resolutions (128×128, 256×256) and video clip lengths (16, 32) to demonstrate the scalability of ECoP. We also consider one audio dataset (Librispeech; Panayotov et al., 2015) and a climate dataset (ERA5; Hersbach et al., 2019) to show the versatility of ECoP.

4.2. Main Experiments

In-domain adaptation. We compare the reconstruction performance of each method on the test datasets across various modalities. We present the results in Table 1, Table 2, and Table 3. Overall, ECoP significantly and consistently outperforms the baselines by a large margin. In particular, measured with PSNR, ECoP outperforms Learnit by 5 dB under the UCF-101 dataset (128×128×16). Moreover, as shown in Figure 3b, ECoP exhibits clear superiority over the baselines in capturing high-frequency components of the signal, including edges in images and dynamic scene changes in videos. We believe this benefit comes from the online context pruning, as it learns to focus more on high-frequency details at the later adaptation steps. We provide more qualitative results in Appendix B.7 and Appendix B.8.

We remark that one of the nice properties of ECoP is the modality- and model-agnosticism: our method covers various modalities from images to climate datasets. On the other hand, we find TransINR struggles to generalize in some modalities and architectures: it requires (a) tokeniza-

Table 2. Reconstruction performance of meta-learned SIREN and NeRV on UCF-101 dataset. We report PSNR (dB), SSIM, and LPIPS where 16 adaptation steps were used for optimization-based meta-learning schemes. N/A denotes the out-of-memory on a single NVIDIA A100 40GB GPU, and the bold indicates the best results of each group.

Resolution	Network	Method	PSNR (↑)	SSIM (↑)	LPIPS (↓)
128×128×16	SIREN (Sitzmann et al., 2020b)	Learnit (Tancik et al., 2021)	25.46	0.720	0.363
		TransINR (Chen & Wang, 2022)	15.14	0.360	0.636
		ECoP (Ours)	**26.59**	**0.769**	**0.237**
	NeRV (Chen et al., 2021)	Learnit (Tancik et al., 2021)	28.86	0.871	0.140
		ECoP (Ours)	**33.99**	**0.949**	**0.019**
256×256×32	SIREN (Sitzmann et al., 2020b)	Learnit (Tancik et al., 2021)	N/A	N/A	N/A
		TransINR (Chen & Wang, 2022)	N/A	N/A	N/A
		ECoP (Ours)	**22.76**	**0.621**	**0.549**
	NeRV (Chen et al., 2021)	Learnit (Tancik et al., 2021)	23.75	0.659	0.422
		ECoP (Ours)	**28.58**	**0.834**	**0.207**

Table 3. PSNR (dB) of meta-learned SIREN on (a) Librispeech, and (b) ERA5 (181×360) datasets. 1 sec contains 16,000 coordinates, and bold indicates the best results of each group.

(a) Librispeech

Method	PSNR (↑) 1 sec	3 sec
Learnit (Tancik et al., 2021)	39.55	31.39
ECoP (Ours)	**43.40**	**36.45**

(b) ERA5

Method	PSNR (↑)
Learnit (Tancik et al., 2021)	64.91
ECoP (Ours)	**74.10**

tion, which is not straightforward for spherical coordinate datasets (e.g., ERA5), and may require the framework to deal with too long sequences of tokens (e.g., videos), and (b) notable modifications when the architecture consists of non-MLP layers (e.g., NeRV) as the framework is specified for MLPs. We also provide the comparison with TransINR utilizing additional test-time optimization in Appendix B.3.

High-resolution signals. One of the significant advantages of ECoP is its exceptional memory efficiency, which allows us to meta-learn on large-scale high-resolution signals. As demonstrated in Table 1 and Table 2, ECoP can even be trained on 256×256×32 resolution videos or 1024×1024 resolution images, which have been impossible to prior work (even under the constraint of NVIDIA A100 40GB GPU) due to their intensive memory usage.

Cross-domain adaptation. We also consider the cross-domain adaptation scenario: we adapt the meta-learned model on different datasets or even different modalities from the meta-training. In particular, we train our method on UCF-101 and adapt to two different image datasets (CelebA and Imagenette) and one video dataset (Kinetics-400; Kay et al., 2017). Table 4 summarizes the results:ECoP significantly improves the performance over the baseline even in this scenario, indicating ECoP has learned a transferable initialization from the diverse motion of UCF-101.

Table 4. Cross-domain reconstruction performance (PSNR; dB) of meta-learned SIREN under UCF-101 (128×128×16) dataset. We adapt the network to a different dataset and modalities.

Modality	Dataset	Method	PSNR (↑)
Image	CelebA (128×128)	Learnit (Tancik et al., 2021)	27.74
		ECoP (Ours)	**28.45**
	Imagenette (128×128)	Learnit (Tancik et al., 2021)	25.18
		ECoP (Ours)	**26.25**
Video	Kinetics-400 (128×128×16)	Learnit (Tancik et al., 2021)	26.42
		ECoP (Ours)	**27.32**

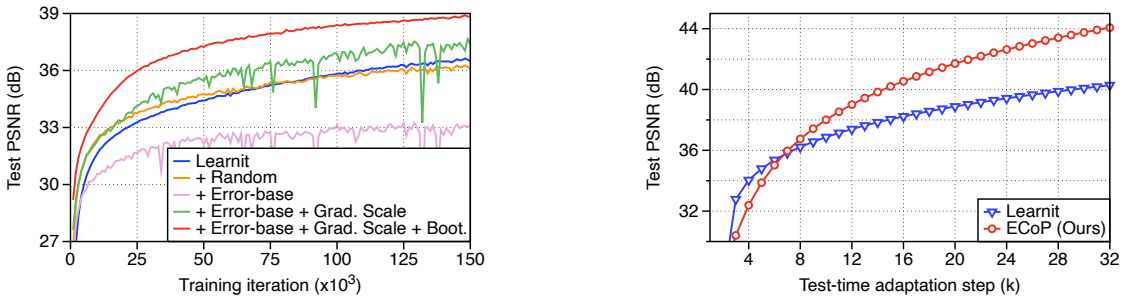
Table 5. Comparison with other efficient meta-learning schemes on SIREN meta-learned on CelebA (178×178) dataset.

Method	PSNR (↑)	SSIM (↑)	LPIPS (↓)
FOMAML (Finn et al., 2017)	25.85	0.669	0.342
Reptile (Nichol et al., 2018)	33.41	0.918	0.084
ECoP (Ours)	**40.54**	**0.975**	**0.005**

Comparison with efficient meta-learning schemes. We also compare ECoP with other efficient optimization-based meta-learning schemes, including first-order MAML (FOMAML; Finn et al., 2017) and Reptile (Nichol et al., 2018). As shown in Table 5, ECoP significantly outperforms the baselines by a large margin. This observation is consistent with prior works (Dupont et al., 2022a), which have also shown that first-order meta-learning schemes tend to struggle with INR learning tasks. Given this, we believe that efficient second-order meta-learning techniques such as ECoP will be a promising direction in this field.

4.3. Ablation Studies

Throughout this section, unless otherwise specified, we perform the experiments on CelebA with SIREN under a smaller batch size for fast training, use the same memory usage for meta-training, and use the same adaptation number at the meta-test stage for all methods. We further



(a) Component analysis (b) Long horizon of ECoP

Figure 4. Test PSNR (dB) of SIREN meta-learned on CelebA (178×178) dataset. (a) Component analysis of ECoP, namely, the use of error-based context pruning (Error-base), gradient scaling (Grad. Scale), and bootstrapped correction (Boot.), and additionally compare with random pruning (Random). (b) Demonstration of the long horizon of ECoP, by reporting PSNR according to the adaptation step k.

Table 6. Comparison of context set choice for generating bootstrapped target on SIREN meta-learned with CelebA (178×178). We consider no bootstrapping (None), random pruning (Random), Error-based pruning (Error-based), and the full context set (Full).

Context set	PSNR (↑)	SSIM (↑)	LPIPS (↓)
None	37.49	0.952	0.017
Random	37.56	0.951	0.016
Error-based	37.59	0.955	0.016
Full (ours)	**38.72**	**0.966**	**0.010**

provide the gradient scaling analysis, coordinate loss analysis, using pruned context set for meta-testing, bootstrapped target's performance during training, and the training-time efficiency of ECoP in Appendix B.

Component analysis. We conduct an in-depth analysis of each training component of ECoP, specifically focusing on the use of (a) error-based context pruning, (b) bootstrapped target, and (c) gradient scaling. We evaluated the performance of these components by comparing the reconstruction performance, as detailed in Figure 4a. Our findings indicate that each component plays a crucial role in improving overall performance. In particular, we find error-based context pruning with gradient scaling itself is quite effective, where it outperforms Learnit and random pruning. This indicates that learning long horizons with an effective sampling strategy indeed helps. Furthermore, we found that incorporating bootstrapped correction not only enhances the overall performance but also stabilizes the training process of error-based sampling, which can be prone to instability when only learning from high-loss samples.

Long horizon of ECoP. Another key benefit of ECoP is that it enables a longer adaptation horizon per-signal (during meta-training) due to the memory saving from the context pruning and thus leads to a better meta-initialization. As shown in Figure 4b, ECoP shows significant improvement as the adaptation step increases (unlike Learnit, which suffers from short-horizon bias), which indicates that our context pruning scheme can be an effective tool for avoiding the myopia of meta-learning when using a large context set.

Bootstrapped target analysis. We performed an experiment to investigate whether the gain of bootstrapped correction mainly comes from the information recovery (by using the full context set) or the longer horizon effect. To this end, we examined various sampling methods for adapting the bootstrapped target, including random context (online), error-based pruned context, and full context set. The results, presented in Table 6, indicate that adaptation using the full context set is indeed effective, and the improvement is attributed to the recovery of information by the target model. Note that, even random and error-based pruned context also improves the performance as it additionally provides more samples but less than the full context set.

5. Discussion and Conclusion

In this paper, we propose an efficient and effective method, ECoP, for fast and scalable implicit neural representation (INR) learning. Here, our key idea is to effectively prune the context set during the meta-training to significantly reduce memory usage while maintaining performance. Our experiments demonstrate that ECoP notably improves the reconstruction performance over various modalities and, more importantly, exhibits superior memory efficiency where it is first to meta-learn on exceptional high-resolution signals.

Future work. While we primarily focus on a case where the context set is used as a target set, i.e., inner and outer loop optimization uses the same set, incorporating ECoP to scenarios where the context and target sets are disjoint will be an interesting future work and worthwhile, e.g., scene rendering. Moreover, extending ECoP for learning extreme high-resolution signals (e.g., long 8K video), where a single forward is not possible under a given memory budget, will be an interesting future direction to explore. We believe that a great variety of techniques can be developed in this direction, for instance, iterative tree-search of high-loss samples by starting from a low-dimension grid and incrementally increasing the resolution of the sampled area through ECoP.

References

Bronskill, J., Massiceti, D., Patacchiola, M., Hofmann, K., Nowozin, S., and Turner, R. Memory efficient meta-learning with large images. In *Advances in Neural Information Processing Systems*, 2021.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. In *IEEE International Conference on Computer Vision*, 2021.

Chen, H., He, B., Wang, H., Ren, Y., Lim, S. N., and Shrivastava, A. NeRV: Neural representations for videos. In *Advances in Neural Information Processing Systems*, 2021.

Chen, Y. and Wang, X. Transformers as meta-learners for implicit neural representations. In *European Conference on Computer Vision*, 2022.

Choi, Y., Uh, Y., Yoo, J., and Ha, J.-W. StarGAN v2: Diverse image synthesis for multiple domains. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

Du, Y., Collins, M. K., , Tenenbaum, B. J., and Sitzmann, V. Learning signal-agnostic manifolds of neural fields. In *Advances in Neural Information Processing Systems*, 2021.

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.

Emam, Z. A. S., Chu, H.-M., Chiang, P.-Y., Czaja, W., Leapman, R., Goldblum, M., and Goldstein, T. Active learning at the imagenet scale. *arXiv preprint arXiv:2111.12880*, 2021.

Feldman, V. and Zhang, C. What neural networks memorize and why: Discovering the long tail via influence estimation. In *Advances in Neural Information Processing Systems*, 2020.

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

Flennerhag, S., Schroecker, Y., Zahavy, T., van Hasselt, H., Silver, D., and Singh, S. Bootstrapped meta-learning. In *International Conference on Learning Representations*, 2022.

Galashov, A., Schwarz, J., Kim, H., Garnelo, M., Saxton, D., Kohli, P., Eslami, S., and Teh, Y. W. Meta-learning surrogate models for sequential decision making. *arXiv preprint arXiv:1903.11907*, 2019.

Garnelo, M., Rosenbaum, D., Maddison, C., Ramalho, T., Saxton, D., Shanahan, M., Teh, Y. W., Rezende, D., and Eslami, S. A. Conditional neural processes. In *International Conference on Machine Learning*, 2018a.

Garnelo, M., Schwarz, J., Rosenbaum, D., Viola, F., Rezende, D. J., Eslami, S., and Teh, Y. W. Neural processes. *arXiv preprint arXiv:1807.01622*, 2018b.

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.

Howard, J. Imagenette, 2019. URL https://github.com/fastai/imagenette/.

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.

Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., Suleyman, M., and Zisserman, A. The kinetics human action video dataset, 2017.

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.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Landgraf, Z., Hornung, A. S., and Cabral, R. S. Pins: Progressive implicit networks for multi-scale neural representations. In *International Conference on Machine Learning*, 2022.

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

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *IEEE International Conference on Computer Vision*, 2015.

Luo, A., Du, Y., Tarr, M. J., Tenenbaum, J. B., Torralba, A., and Gan, C. Learning neural acoustic fields. In *Advances in Neural Information Processing Systems*, 2022.

Martel, J. N., Lindell, D. B., Lin, C. Z., Chan, E. R., Monteiro, M., and Wetzstein, G. Acorn: Adaptive coordinate networks for neural scene representation. *ACM Trans. Graph. (SIGGRAPH)*, 2021.

Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., and Geiger, A. Occupancy networks: Learning 3D reconstruction in function space. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

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.

Müller, T., Evans, A., Schied, C., and Keller, A. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph. (SIGGRAPH)*, 2022.

Nichol, A., Achiam, J., and Schulman, J. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

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.

Paul, M., Ganguli, S., and Dziugaite, G. K. Deep learning on a data diet: Finding important examples early in training. In *Advances in Neural Information Processing Systems*, 2021.

Rajeswaran, A., Finn, C., Kakade, S. M., and Levine, S. Meta-learning with implicit gradients. *Advances in Neural Information Processing Systems*, 2019.

Rolnick, D., Ahuja, A., Schwarz, J., Lillicrap, T., and Wayne, G. Experience replay for continual learning. In *Advances in Neural Information Processing Systems*, 2019.

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

Sener, O. and Savarese, S. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

Shin, J., Lee, H. B., Gong, B., and Hwang, S. J. Large-scale meta-learning with continual trajectory shifting. In *International Conference on Machine Learning*, 2021.

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., Chan, E. R., Tucker, R., Snavely, N., and Wetzstein, G. MetaSDF: Meta-learning signed distance functions. In *Advances in Neural Information Processing Systems*, 2020a.

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*, 2020b.

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

Snell, J., Swersky, K., and Zemel, R. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, 2017.

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.

Sorscher, B., Geirhos, R., Shekhar, S., Ganguli, S., and Morcos, A. S. Beyond neural scaling laws: beating power law scaling via data pruning. In *Advances in Neural Information Processing Systems*, 2022.

Tack, J., Park, J., Lee, H., Lee, J., and Shin, J. Meta-learning with self-improving momentum target. In *Advances in Neural Information Processing Systems*, 2022.

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.

Tian, Y., Krishnan, D., and Isola, P. Contrastive multiview coding. In *European Conference on Computer Vision*, 2020.

Titsias, M. K., Schwarz, J., Matthews, A. G. d. G., Pascanu, R., and Teh, Y. W. Functional regularisation for continual learning with gaussian processes. In *International Conference on Learning Representations*, 2020.

Toneva, M., Sordoni, A., Combes, R. T. d., Trischler, A., Bengio, Y., and Gordon, G. J. An empirical study of example forgetting during deep neural network learning. In *International Conference on Learning Representations*, 2019.

Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing (TIP)*, 2004.

Xie, S., Sun, C., Huang, J., Tu, Z., and Murphy, K. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *European Conference on Computer Vision*, 2018.

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.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Appendix

A. Experimental Details

In this section, we describe the experimental details of Section 4, including ECoP and the baselines. We also provide the implementation of ECoP in the supplementary material.

A.1. Dataset Details

CelebA. CelebA is a fine-grained dataset that consists of the face image of celebrities (Liu et al., 2015). The dataset comprises 202,599 images, where we use 162K for training, 20K for validation, and 20K for testing. We resize the image into 178 and then apply a center crop of 178, which ends with a resolution of 178×178. We pre-process pixel coordinates into $[-1, 1]^2$ and feature values ranging from 0 to 1.

Imagenette. Imagenette is a 10-class subset of the ImageNet (Deng et al., 2009) dataset, which is comprised of 9,000 training and 4,000 test images (Howard, 2019). We resize the image into 178 and then apply a center crop of 178, which ends with a resolution of 178×178. We pre-process pixel coordinates into $[-1, 1]^2$ and feature values ranging from 0 to 1.

Text. Text dataset consists of text image with a resolution of 178×178 (Tancik et al., 2021). We pre-process the pixel coordinates into $[-1, 1]^2$ and feature values ranging from 0 to 1.

ImageNet-100. ImageNet-100 is a random subset of ImageNet (Deng et al., 2009) data, which includes 100 classes (Tian et al., 2020). We resize the image into 256, then center crop the image to get 256×256 resolution image. We pre-process the pixel coordinates into $[-1, 1]^2$ and feature values ranging from 0 to 1.

CelebA-HQ. CelebA-HQ is a high-resolution fine-grained dataset, which includes images of celebrities (Karras et al., 2018). We divided the dataset into 27,000 training and 3,000 test samples and pre-processed the pixel coordinates into $[-1, 1]^2$ and feature values ranging from 0 to 1. We consider two resolutions, i.e., 512×512 and 1024×1024.

AFHQ. AFHQ is a high-resolution fine-grained dataset, which includes animal faces consisting of 15,000 images at 512×512 resolution (Choi et al., 2020). We divided the dataset into 14,336 training and 1,467 testing points, and pre-processed the pixel coordinates into $[-1, 1]^2$ and feature values ranging from 0 to 1.

UCF-101. UCF-101 is a video dataset comprising 13,320 videos (9,357 training and 3,963 test videos) with a resolution of 320×240, where the action classification consists of 101 classes (Soomro et al., 2012). Each video clip is center-cropped to 240×240 and then resized into 128×128 and 256×256 with a video clip length of 16 and 32, respectively. We pre-process the pixel coordinates into $[-1, 1]^3$ and feature values ranging from 0 to 1.

Kinetics-400. For the cross-domain adaptation purpose, we use the mini-Kinetics-200 dataset (Xie et al., 2018), which consists of 200 categories with the most training samples from the Kinetics-400 dataset (Kay et al., 2017). We center-crop the videos to the same height and width and then resize them into 128×128 resolution, and use the frame of 16 clips. We pre-process the pixel coordinates into $[-1, 1]^3$ and feature values ranging from 0 to 1.

ERA5. ERA5 is a dataset comprised of temperature observations on a global grid of equally spaced latitudes and longitudes (Hersbach et al., 2019). By following (Dupont et al., 2022a), we use the grid resolution of 181×360 by resizing the grid. We interpret each time step as an independent signal, and the dataset comprises 9,676 training points and 2,420 test points. For the input pre-processing, given latitudes ρ and longitudes φ are transformed into 3D Cartesian coordinates $\mathbf{c} = (\cos\rho \cos\varphi, \cos\rho \sin\varphi, \sin\rho)$ where latitudes ρ are equally spaced between $-\frac{\pi}{2}$ and $\frac{\pi}{2}$ and longitudes φ are equally spaced between 0 and $\frac{2\pi(n-1)}{n}$ where n the number of distinct values of longitude (360).

LibriSpeech. LibriSpeech is an English speech recording collection at a 16kHz sampling rate (Panayotov et al., 2015). By following Dupont et al. (2022b), we use the train-clean-100 split for training and test-clean split for testing, i.e., 28,539 training and 2,620 test samples. For the main experiments, we use the first 1 second and 3 seconds of each example (1 second contains 16,000 coordinates). For the pre-processing, we scale the coordinates into $[-50, 50]$.

A.2. Training and Evaluation Details

Network architecture details. For the main experiment, we mainly use SIREN, a multi-layer perception (MLP) with sinusoidal activation functions (Sitzmann et al., 2020b), i.e., $\mathbf{x} \mapsto \sin(\omega_0(\mathbf{W}\mathbf{x} + \mathbf{b}))$ where $\mathbf{W}, \mathbf{b}$ are weight and biases of the MLP layer and ω_0 is the fixed hyperparameter. For image, audio, and manifold datasets, we use SIREN with 5 layers with 256 hidden dimensions, and for video, we use 7 layers with the same hidden dimension. We used $\omega_0 = 50$ for the manifold dataset and use $\omega_0 = 30$ for the rest. We additionally consider NeRV (Chen et al., 2021) for the video dataset. For the UCF-101 dataset of 128×128×16, we use 4 NeRV blocks, and for 256×256×32, we use 5 NeRV blocks.

Training details. For all dataset, we use Adam optimizer (Kingma & Ba, 2015) for the outer loop. We use the outer step of 150,000, except for learning the UCF-101 dataset with SIREN, where we use 100,000 steps. For SIREN, we use the outer learning rate of $\beta = 3.0 \times 10^{-6}$ for Librispeech and use $\beta = 1.0 \times 10^{-5}$ for the rest. For NeRV, we use the outer learning rate of $\beta = 1.0 \times 10^{-4}$. As for the inner loop learning rate, we use $\alpha = 1.0 \times 10^{-2}$, and $\alpha = 1.0 \times 10^{-1}$ for SIREN and NeRV, respectively. For inner step number K, ECoP is trained on a longer horizon than Learnit by multiplying $1/\gamma$ (which uses the same memory usage). For Learnit, we use $K = 4$ for most of the dataset, where we use $K = 1$ for CelebA-HQ (1024×1024), $K = 2$ for UCF-101 (128×128×16) on SIREN, $K = 5$ on UCF-101 (256×256×32) on NeRV, and $K = 20$ on UCF-101 (128×128×16) on NeRV. We use the same batch size for Learnit and ECoP to fairly use the memory, e.g., ECoP use $K = 16$ for CelebA as we use the data ratio $\gamma = 0.25$ for CelebA. The batch size was selected differently across the dataset (e.g., under the GPU memory budget), where we found the increased batch size slightly improved the performance and the stability, but it did not have a significant effect.

Hyperparameter details for ECoP. We find that the hyperparameter introduced by ECoP is not sensitive across datasets and architectures. For the sampling ratio γ, i.e., the ratio of retaining coordinates, we use 0.25 for most of the dataset except for Librispeech and ERA5, where we used 0.5 (as we do not need to prune the context much for these low-resolution signals), and $0.5, 0.2$ when training NeRV on UCF-101 on 128 and 256 resolution, respectively. For the bootstrapped correction hyperparameters, we used $L = 5$, and $\lambda = 100$, where we believe tuning these hyperparameters will indeed improve the performance much more (we did not tune extensively).

Evaluation details. For the evaluation, to fairly compare with the baseline, we use the same test-time adaptation step for Learnit and ECoP. Here, we use the same step number that is used by ECoP on meta-training. For the gradient scaling, we use the same sampling ratio γ to compute the high-loss sample. We additionally compare the results of test-time adaptation of TransINR in Section B.3.

Resource details. For the main development, we mainly use Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz and a single RTX 3090 24GB GPU, except for high-resolution signals (e.g., CelebA-HQ of 1024×1024) where we use AMD EPYC 7542 32 Core Processor and a single NVIDIA A100 SXM4 40GB.

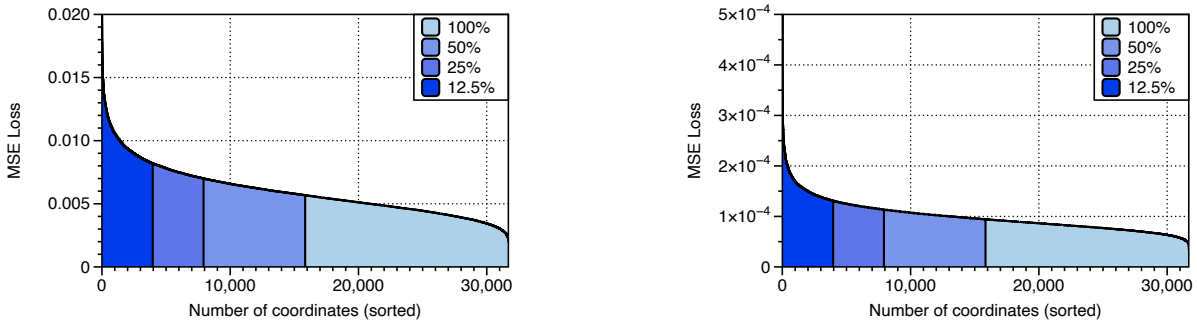
A.3. Baseline Details

In this section, we explain the meta-learning baselines we used for evaluating ECoP at a high level.

- **Learnit** (Tancik et al., 2021) utilizes the second-order gradient version of model-agnostic meta-learning (MAML; Finn et al., 2017) for learning INRs.
- **TransINR** (Chen & Wang, 2022) utilizes Transformer as a meta-learner to predict the INR parameter with a given context set, and additionally proposed a parameter-efficient INR architecture specialized for MLP, i.e., weight grouping.
- **FOMAML** (Finn et al., 2017) is a memory-efficient MAML that utilizes first-order gradients for the inner loop update, i.e., no need to save the inner loop adaptation gradients when calculating the outer loop loss.
- **Reptile** (Nichol et al., 2018) is a memory-efficient meta-learning that utilizes first-order gradients for the inner loop and uses the parameter difference between the meta-initialization and the adapted model to calculate the outer loop loss.

B. More Experimental Results

B.1. Loss Statistic of Coordinates



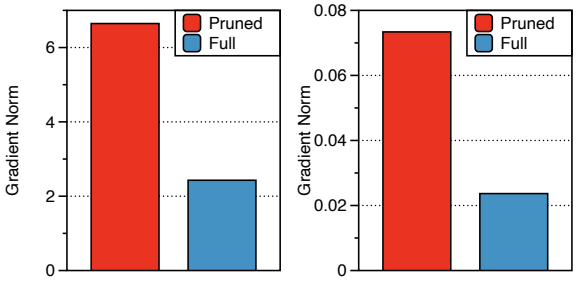
(a) $k = 0$ (b) $k = 15$

Figure 5. Loss statistic of coordinates where the indexes are sorted with the loss value and the highlighted region indicates the pruned context set (with error-based) by a given sampling ratio hyper-parameter $\gamma \times 100$ (%). k indicates the adaptation step number. We meta-learn SIREN on CelebA (178×178) dataset with ECoP.

To understand the behavior of ECoP , we analyze the loss statistics of the coordinates. Here, we visualize the error of the given context set $\mathcal{C}_{\texttt{full}}$ at adaptation iteration k, namely $\{R_k(\mathbf{x}, \mathbf{y}) | (\mathbf{x}, \mathbf{y}) \in \mathcal{C}_{\texttt{full}}\}$. Figure 5 shows the loss statistics where the indexes are sorted by the loss value. As shown in the figure, the distribution is quite similar to the Pareto distribution, where sampling the high-loss values can be representative points over the context set.

B.2. Analysis on Gradient Scaling

To further investigate the impact of gradient scaling, we analyze the gradient norms during the adaptation phase of meta-training. For this experiment, we train SIREN with ECoP on the CelebA dataset. Here, we plot the gradient norms at the adaptation step k measured by the full context set $\mathcal{C}_{\texttt{full}}$, and the error-based pruned context set $\mathcal{C}^k_{\texttt{high}}$, respectively. As shown in Figure 6, our results indicate that the norm of the gradients exhibits significant variations, with some steps exhibiting 3× larger gradients when using the pruned context set. This suggests that the meta-learner employed a larger step size during training, which highlights the importance of test-time gradient scaling. We also find that scaling the gradient with the loss ratio of the full context set and the sampled context leads to similar performance improvements and may serve as a faster alternative, as it eliminates the need for gradient calculation twice, i.e., showing 40.21 in PSNR where the gradient scaling with gradient norms shows 40.54.



(a) $k = 0$ (b) $k = 15$

Figure 6. Gradient norm of the full context set $\mathcal{C}_{\texttt{full}}$ and the error-based pruned context set $\mathcal{C}^k_{\texttt{high}}$ at iteration k.

B.3. Additional Comparison with TransINR under Test-time Optimization

Table 7. Comparison with TransINR under same test-time adaptation steps on SIREN meta-learned with CelebA (178×178) dataset. We further adapt the same adaptation steps (as ECoP) from the predicted network from TransINR.

Method	PSNR (↑)	SSIM (↑)	LPIPS (↓)
TransINR (Chen & Wang, 2022)	32.37	0.913	0.068
TransINR (Chen & Wang, 2022) + Test-time optimization	34.12	0.932	0.046
ECoP (Ours)	**40.54**	**0.975**	**0.005**

To further verify the superiority of ECoP, we additionally compare with the test-time optimization performance of TransINR. Here, we use SGD with the same adaptation steps (as ECoP) to further optimize the predicted INR by TransINR. As shown in Table 7, ECoP significantly shows better results even under this scenario. Still, note that such a comparison is not fair for ECoP, as TransINR additionally uses a big Transformer encoder over ECoP, which is quite computationally expensive.

B.4. Performance of Bootstrapped Target during Meta-training

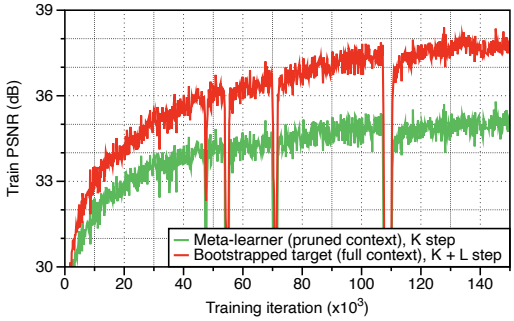


Figure 7. Meta-training reconstruction performance (PSNR;dB) of bootstrapped target model $\theta^{\texttt{boot}}_{K+L}$ and the meta-learner with K step adaptation θ_K on CelebA (178×178) dataset. Note that the meta-learner uses the pruned context set for K steps, hence, shows comparably lower PSNR than the meta-test time.

To understand how the bootstrapped target improves the performance of the meta-learner, we compare the performance of bootstrapped target $\theta^{\texttt{boot}}_{K+L}$ and the meta-learner θ_K (adapted K steps from the meta-initialization with the error-based pruned context set) during the meta-training stage. As shown in Figure 7, we observe that the performance of bootstrapped target is consistently better than the meta-learner with K step adaptation, which indicates that the proposed target indeed helps for learning a better initialization. Such observation is quite similar to the prior works in meta-learning (Tack et al., 2022) and self-supervised learning (Caron et al., 2021), where a consistently better-performing teacher can improve the student model's generalization.

B.5. Effectiveness of Using Full Context Set for Meta-testing

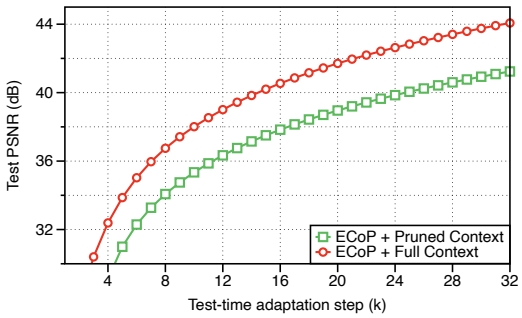


Figure 8. Comparison of test reconstruction performance (PSNR; dB) between the utilization of full context set and error-based pruned context set during meta-test. The experiment is conducted over a meta-learn SIREN on CelebA (178×178) dataset with ECoP.

To verify that the utilization of a full context set for meta-testing is truly effective, we compare the adaptation performance when using full and the error-based pruned context set. Here, we only apply the gradient scaling when using the full context set, as pruned context set gradient norm mismatch does not occur between the meta-training and testing. As shown in Figure 8, using the full context set for the meta-test time is indeed effective and shows consistent improvement over the pruned context set. Note that using pruned context set is also quite effective as it shows better adaptation performance than Learnit on a long adaptation horizon.

B.6. Training Time Efficiency of ECoP

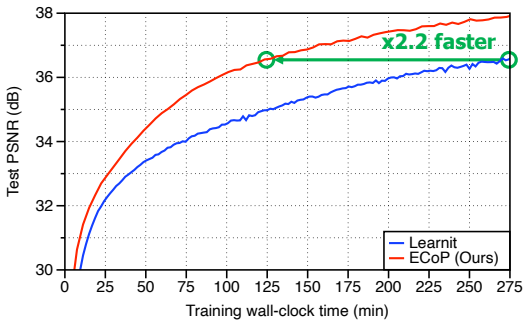


Figure 9. Comparison of test reconstruction performance (PSNR; dB) between Learnit and ECoP, under the same training wall-clock time. The experiment is conducted over a meta-learn SIREN on CelebA (178×178) dataset.

We find that ECoP is also efficient in terms of training time. Our method may be seemingly compute-inefficient in terms of training time as it additionally requires bootstrapped target generation and utilizes a longer horizon adaptation step, however, we show that it is not. Although ECoP increases the training time of Learnit by roughly 2 times when we use (i) 4 times longer adaptation step and (ii) additional bootstrapped correction (which overall uses the same memory as Learnit with $\gamma = 0.25$), we have observed it is that it is more than 2 times faster to achieve the best performance of Learnit: in Figure 9, we compare the accuracy under the same training wall-clock time with Learnit. Furthermore, one can easily reduce the training time of ECoP by reducing the adaptation step number, which can even bring significant memory efficiency.

B.7. More Qualitative Comparison with Baselines on Image datasets

Figure 10. Qualitative comparison between ECoP and baselines on high-resolution AFHQ (512×512) dataset.

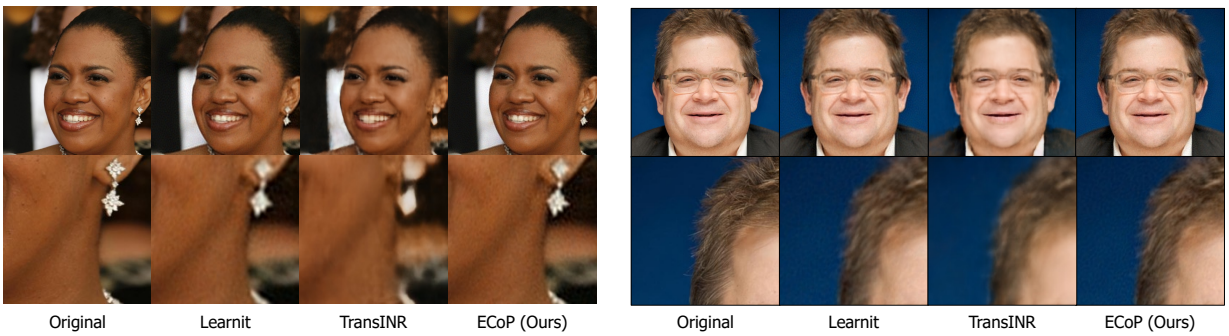
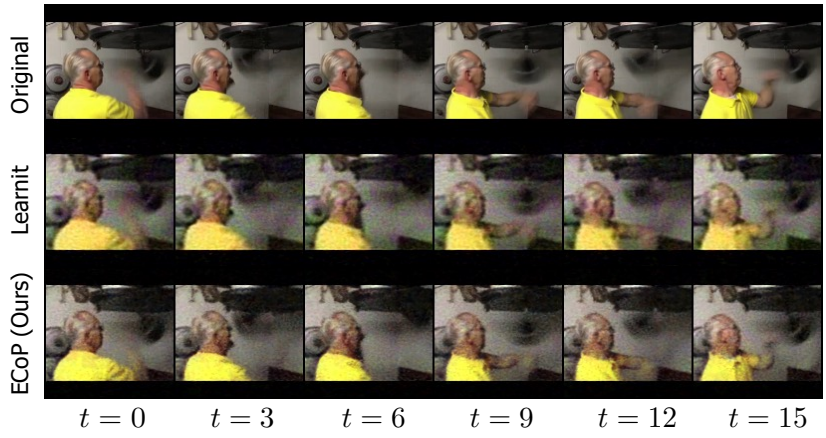
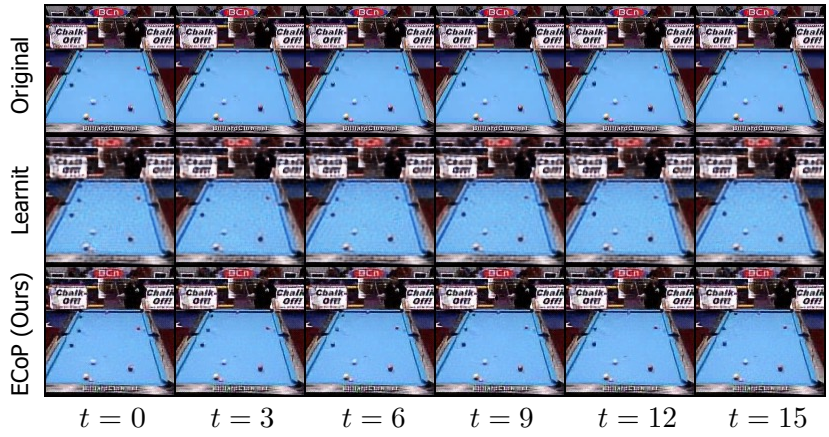


Figure 11. Qualitative comparison between ECoP and baselines on high-resolution CelebA-HQ (512×512) dataset.

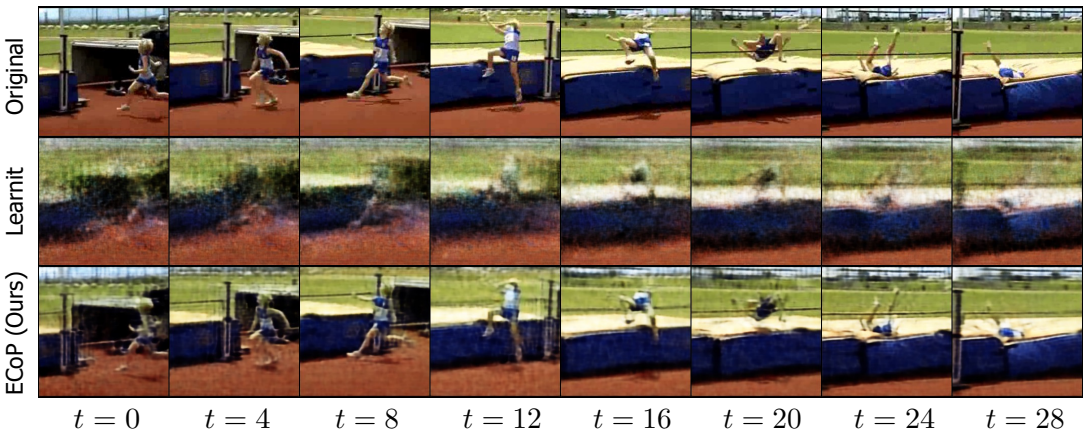
B.8. More Qualitative Comparison with Baselines on the Video dataset



(a) UCF-101 (128×128×16) on SIREN



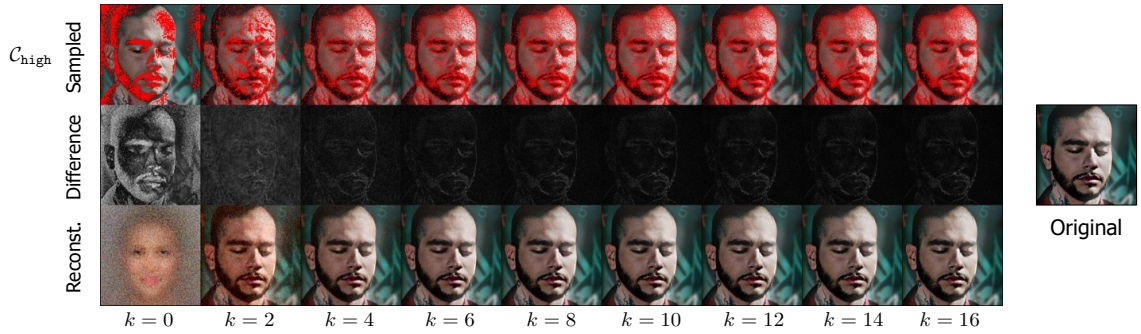
(b) UCF-101 (128×128×16) on NeRV



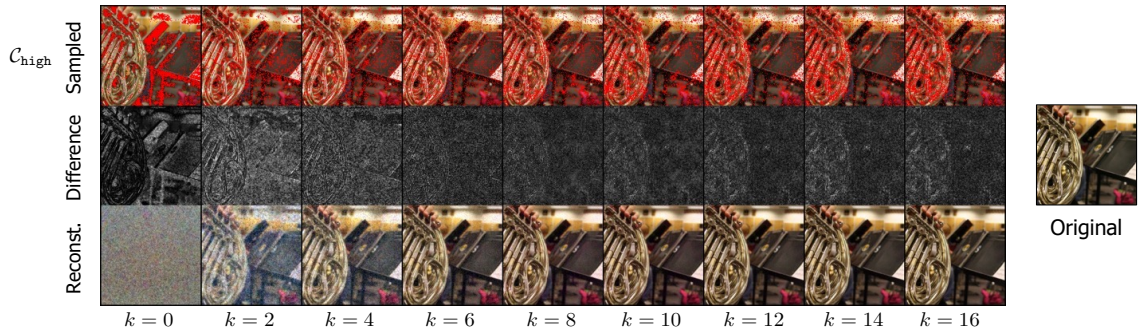
(c) UCF-101 (256×256×32) on NeRV

Figure 12. Qualitative comparison between ECoP and Learnit on UCF-101 dataset.

C. More Visualizations of Sampled Context Points via ECoP on Image datasets

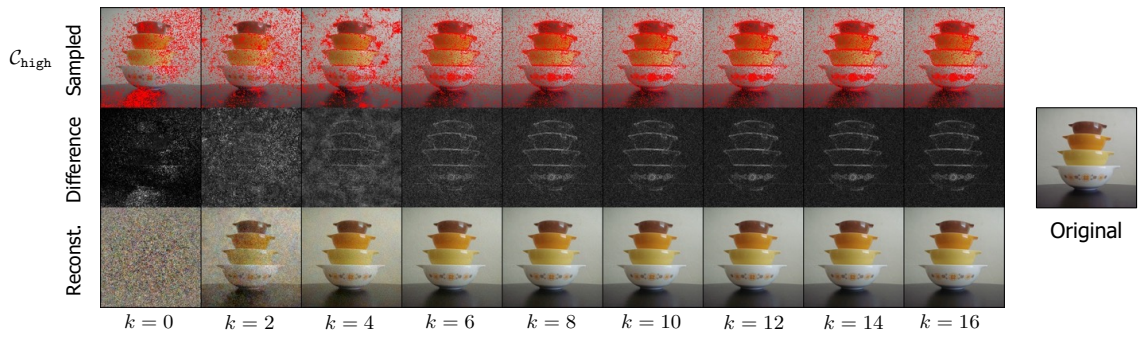


(a) CelebA-HQ



(b) Imagenette

(c) Text



(d) ImageNet-100

Figure 13. Visualization of sampled points (first), the difference between the original signal (middle), and the reconstructed signal (last) via ECoP trained on (a) CelebA-HQ, (b) Imagenette, (c) Text, and (4) ImageNet-100 with SIREN. The sampled coordinates are highlighted in red where the sampling ratio γ is 0.25, and k denotes the adaptation step.

D. More Visualizations of Sampled Context Points via ECoP on the Video dataset

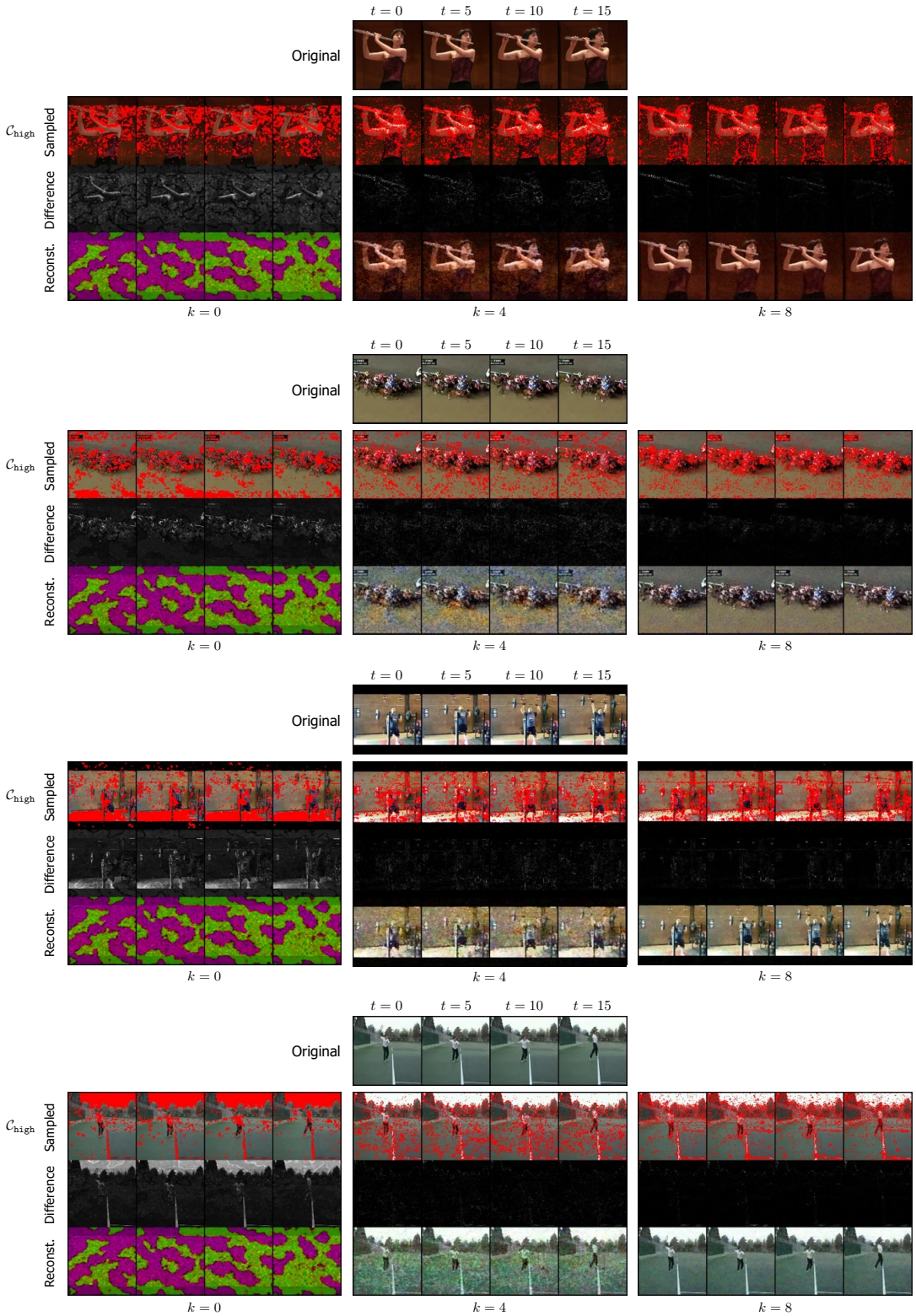


Figure 14. Visualization of sampled points (first), the difference between the original signal (middle), and the reconstructed signal (last) via ECoP trained on UCF-101 with SIREN. The sampled coordinates are highlighted in red where the sampling ratio γ is 0.25, k denotes the adaptation step, and t denotes the frame index in each video sequence.