Learning Large-scale Neural Fields via Context Pruned Meta-Learning

Jihoon Tack, Subin Kim, Sihyun Yu, Jaeho Lee, Jinwoo Shin, Jonathan Richard Schwarz

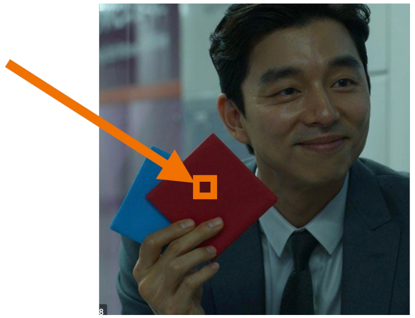
NEURAL INFORMATION PROCESSING SYSTEMS
KAIST
POSTECH
POHANG UNIVERSITY OF SCIENCE AND TECHNOLOGY
UCL

Neural Fields (NFs) / Implicit Neural Representations (INRs)

NFs represents signals as continuous coordinate-mapping functions parameterized by a neural net

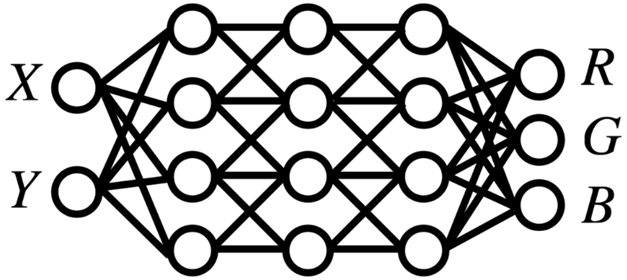
For instance, an **RGB** **image** can be represented as a function taking a form

$$\begin{array}{rcl} \texttt{coords} & \longrightarrow & \texttt{values} \\ (X, Y) & \longrightarrow & (R, G, B) \\ \mathbb{R}^2 & \longrightarrow & \mathbb{R}^3 \end{array}$$



$$f_{\text{img}}\left(\frac{76}{255}, \frac{152}{255}\right) = (0.95, 0.03, 0.04)$$

This can be learned by a neural network (usually a MLP) with $d_{\text{in}} = 2, d_{\text{out}} = 3$



$$\min_{\theta} \left\| f_{\text{img}} - f_{\theta} \right\|_{L_2}$$

Why are Neural Fields interesting?

Neural Fields have the **potential** to be a popular form of **data representation** in the near future!

- Interesting point 1. Effective at novel view synthesis (or resolution free!) [1,2]

Pixels Bilinear INR-decoder

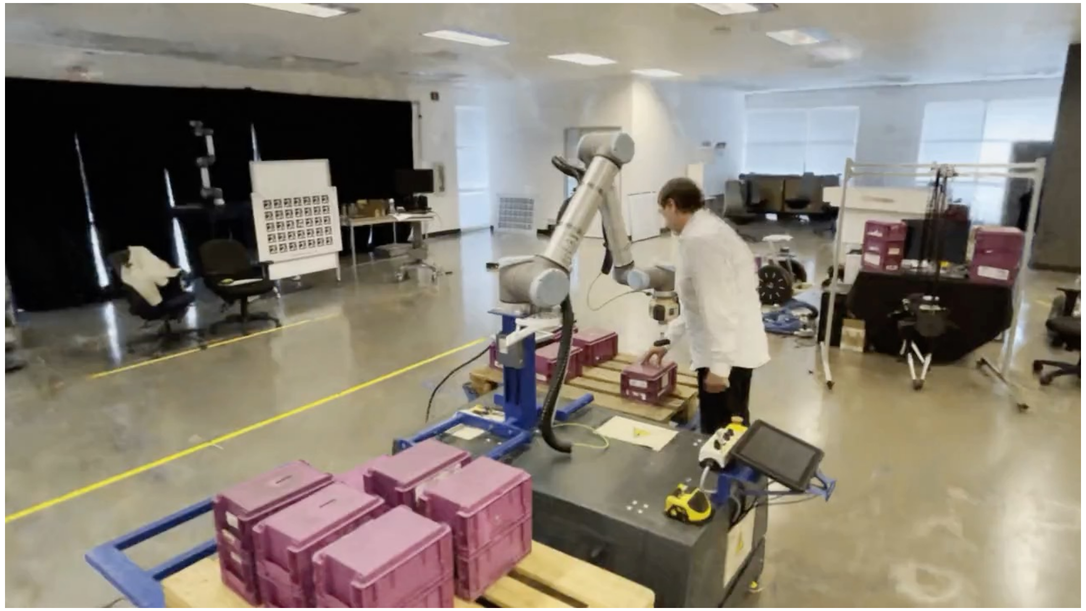
Rendering: Image → 3D

[1] Chen et al. Learning Continuous Image Representation with Local Implicit Image Function. CVPR 2021
[2] Mildenhall et al. NeRF in the Dark: High Dynamic Range View Synthesis from Noisy Raw Images. CVPR 2022

Why are Neural Fields interesting?

Neural Fields have the **potential** to be a popular form of **data representation** in the near future!

- Interesting point 2. Represent complex signals [3,4], e.g., large-scale 3d scenes, videos

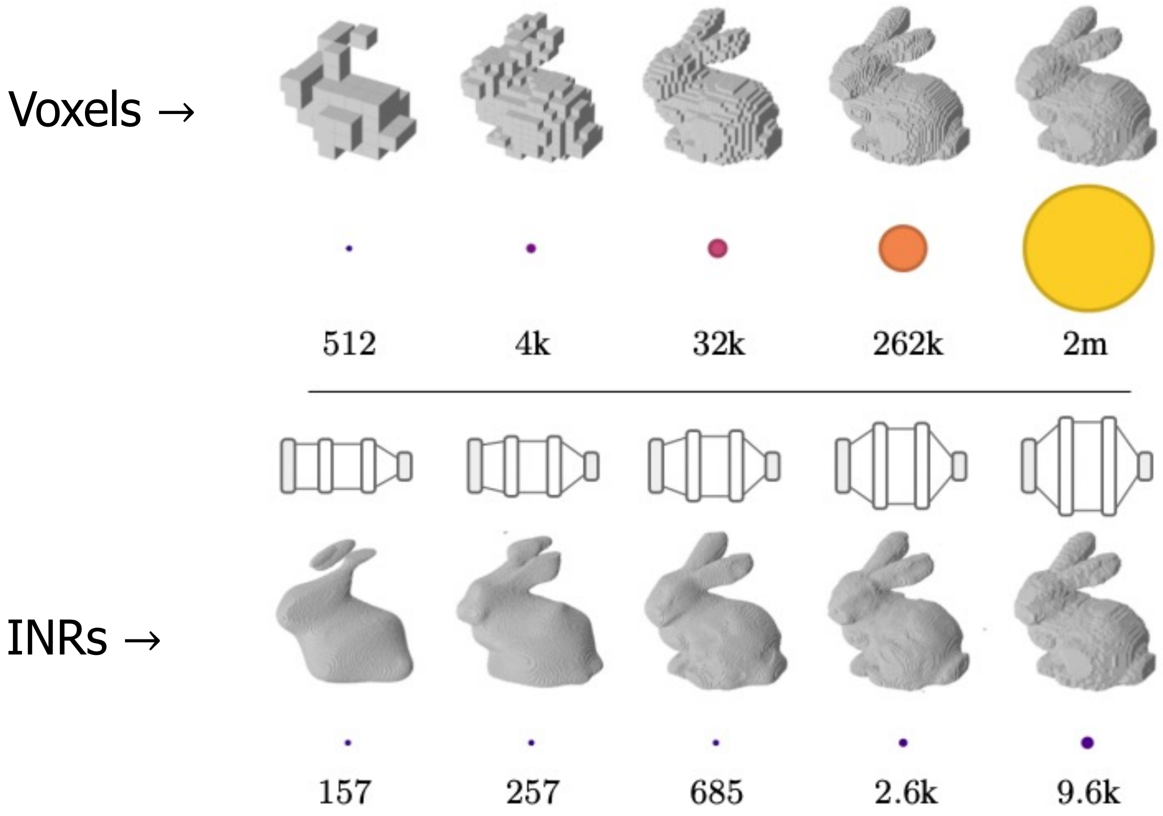


[3] Tancik et al. Block-NeRF: Scalable Large Scene Neural View Synthesis. CVPR 2022
[4] Muller et al. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. SIGGRAPH 2022

Why are Neural Fields interesting?

Neural Fields have the **potential** to be a popular form of **data representation** in the near future!

- Interesting point 3. Storage efficient [5]

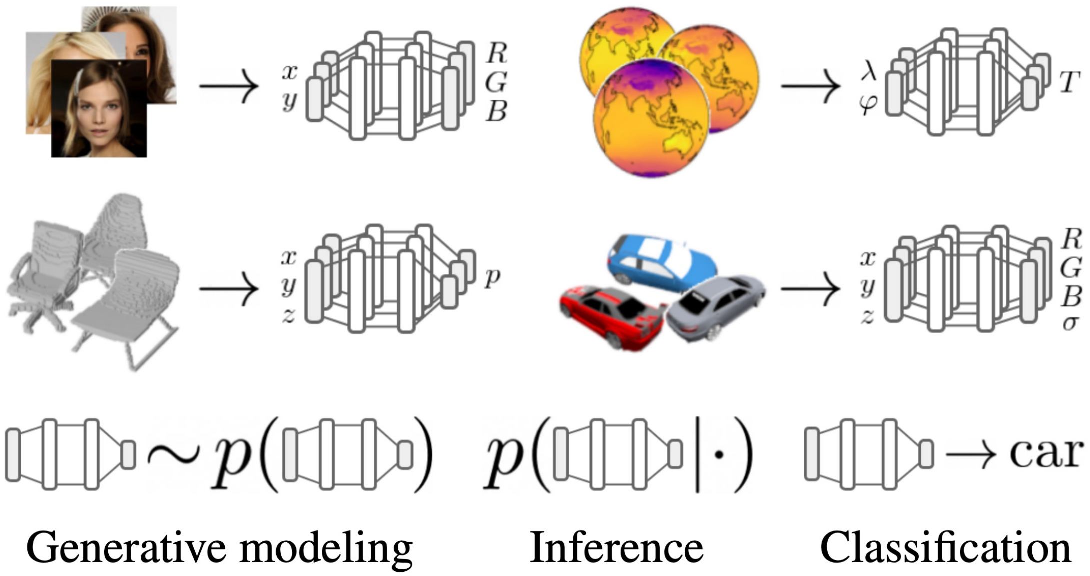


[5] Dupont et al. From data to functa: Your data point is a function and you should treat it like one. ICML 2022

Why are Neural Fields interesting?

Neural Fields have the **potential** to be a popular form of **data representation** in the near future!

- Interesting point 4. NF it-self can be used as a data point! [6,7]



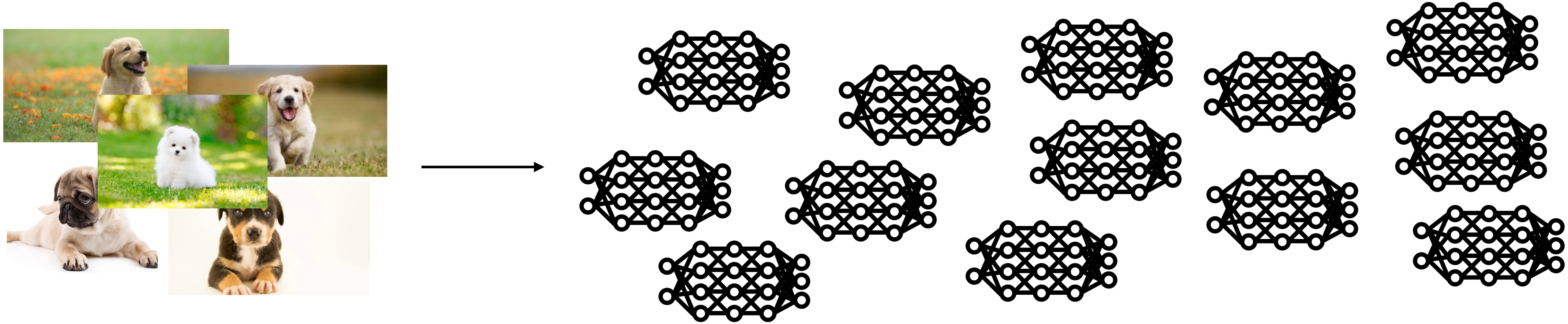
Step 1: Fit data points into INRs

Step 2: Run downstream tasks, e.g., classification

[6] Dupont et al. From data to functa: Your data point is a function and you should treat it like one. ICML 2022
[7] Bauer et al., Spatial Functa: Scaling Functa to ImageNet Classification and Generation. ICLR workshop on Neural Fields 2023

An Obstacle

Q. Can we scale such an idea to a **Big Dataset**? / How to train a **foundation model** for NFs?



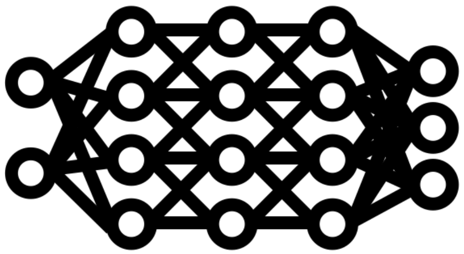
Imagine training **ImageNet number** of these....

An Obstacle

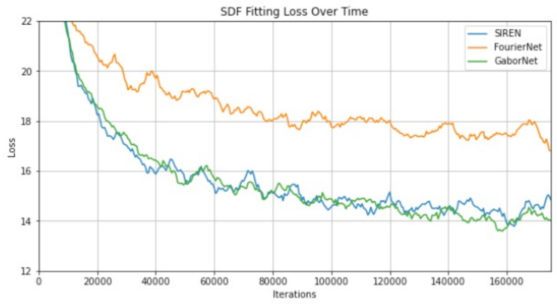
Q. Can we scale such an idea to a **Big Dataset**? / How to train a **foundation model** for NFs?

We mainly face **three challenges**

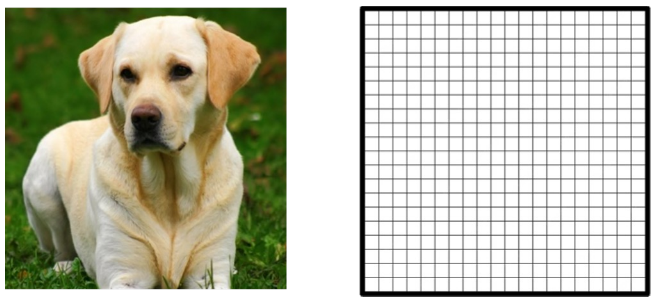
- how to fit neural fields in a **parameter-efficient** way
- how to fit neural fields in a **time-efficient** way
- how to fit neural fields in a **memory-efficient** way



Need to save a neural network…



Training time is costly…



Memory issue when learning high-resolution signals…

An Obstacle

Q. Can we scale such an idea to a **Big Dataset**? / How to train a **foundation model** for NFs?

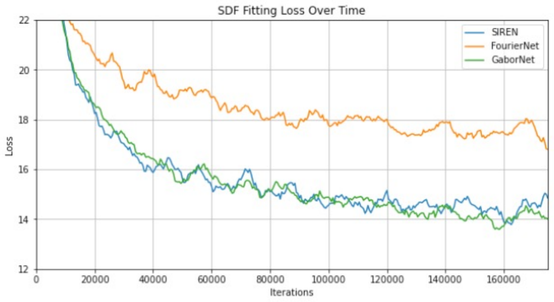
We mainly face **three challenges**

- how to fit neural fields in a **parameter-efficient** way
- how to fit neural fields in a **time-efficient** way
- how to fit neural fields in a **memory-efficient** way

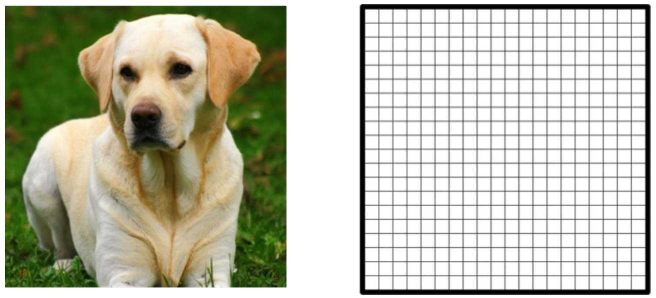
GradNCP (this paper)
→ time and memory-efficient learning for neural fields

Time and parameter-efficient learning for neural fields?
→ *Check our prior works* [8,9]

Need to save a neural network…



Training time is costly…



Memory issue when learning high-resolution signals…

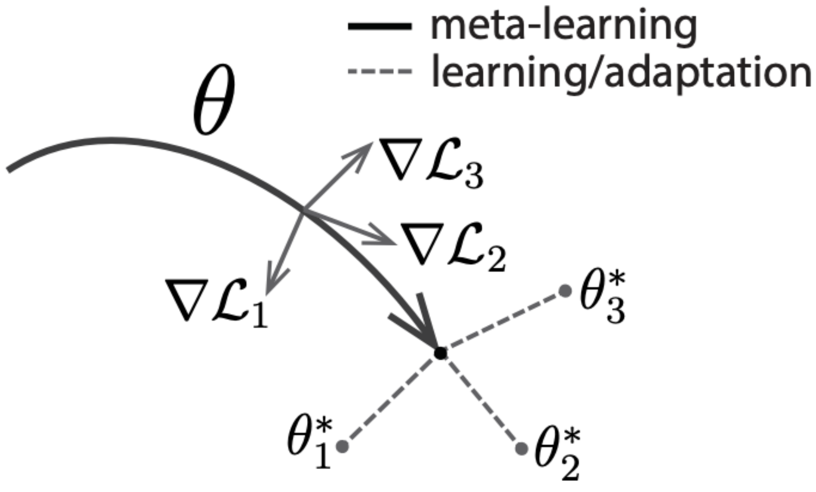
[8] Lee et al. Meta-Learning Sparse Implicit Neural Representations. NeurIPS 2021
[9] Schwarz et al., Modality-Agnostic Variational Compression of Implicit Neural Representations. ICML 2023

Time-efficiency: Use Meta-Learning

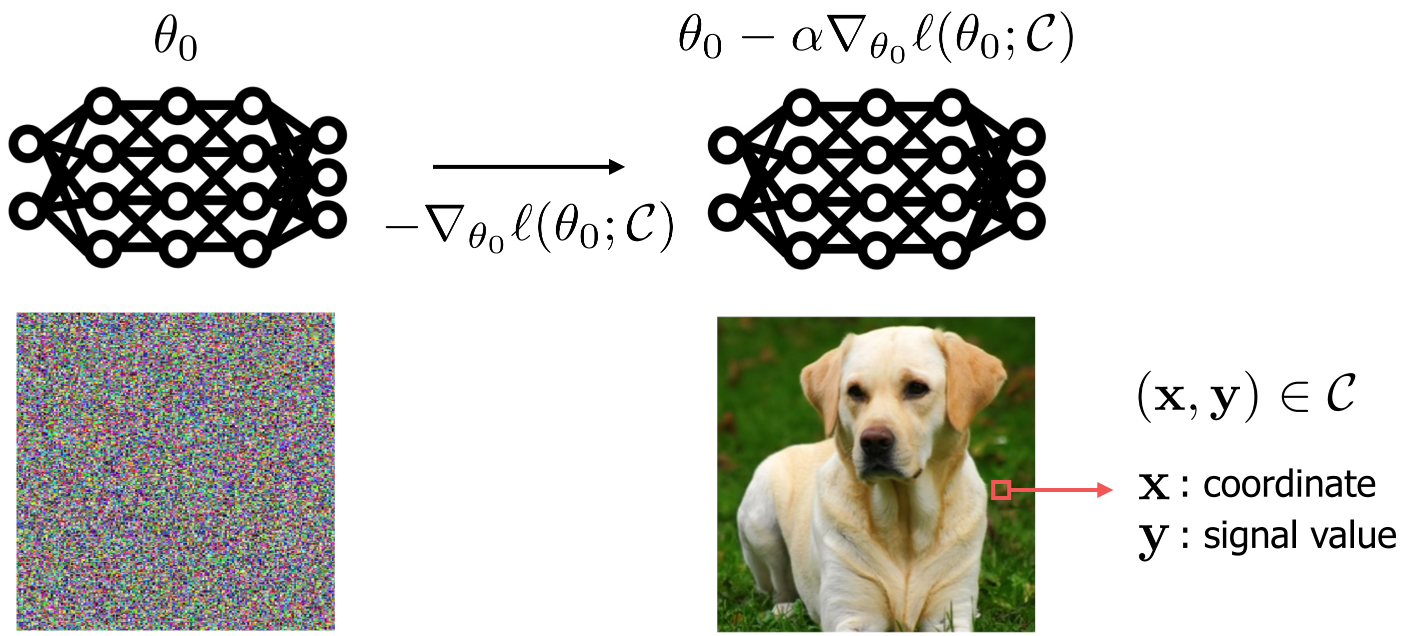
Prior works have use meta-learning; **optimization-based meta-learning** shows versatile usages

→ learning a good initialization [10]

Objective?
Find an initialization θ_0 such that **few-step gradients** can fit the signal



MAML



[10] Finn et al. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks, ICML 2017

Time-efficiency: Use Meta-Learning

Prior works have use meta-learning; **optimization-based meta-learning** shows versatile usages

→ learning a good initialization [10]

Objective?
Find an initialization θ_0 such that **few-step gradients** can fit the signal

MAML

Train over multiple signals (batch of signals):

$$\theta_0 = \mathrm{argmin}_{\theta_0} \frac{1}{N}\sum_{i=1}^{N} \ell(\theta_0 - \boxed{\alpha\nabla_{\theta_0}\ell(\theta_0;\mathcal{C}^{(i)})};\mathcal{C}^{(i)})$$

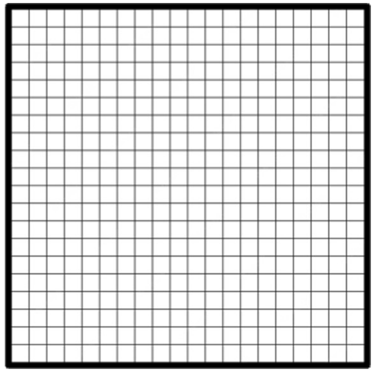
Hessian computation: **memory scales linearly with the number of context set**

[10] Finn et al. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks, ICML 2017

Time-efficiency: Use Meta-Learning

As the signal resolution increases *memory usage rapidly increases*

- Image with resolution 224 × 224? **Context set size of 50,176** (# of input coordinates)



For videos…? we should consider the timestep!
→ e.g., 50,176 * 16 (about 800K)

θ_1^* θ_2^*

MAML

Train over multiple signals (batch of signals):

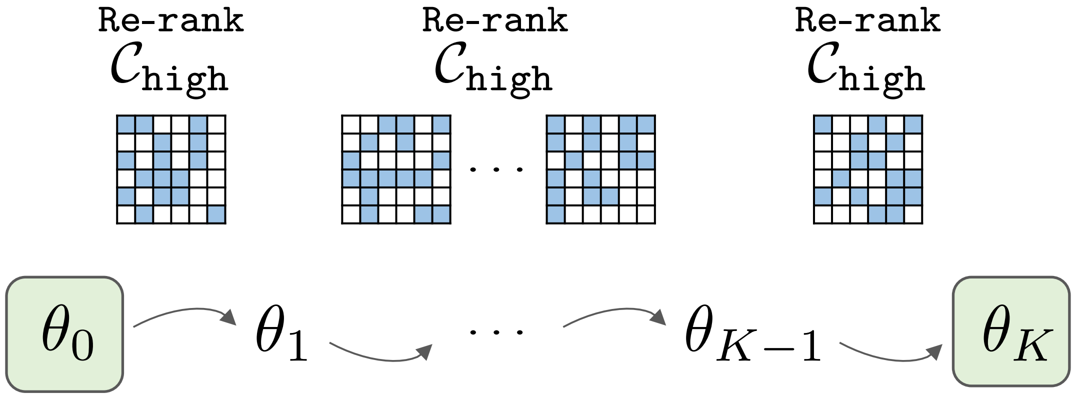
$$\theta_0 = \operatorname{argmin}_{\theta_0} \frac{1}{N} \sum_{i=1}^{N} \ell(\theta_0 - \boxed{\alpha \nabla_{\theta_0} \ell(\theta_0; \mathcal{C}^{(i)})}; \mathcal{C}^{(i)})$$

Hessian computation: **memory scales linearly with the number of context set**

[10] Finn et al. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks, ICML 2017

High-level overview of GradNCP

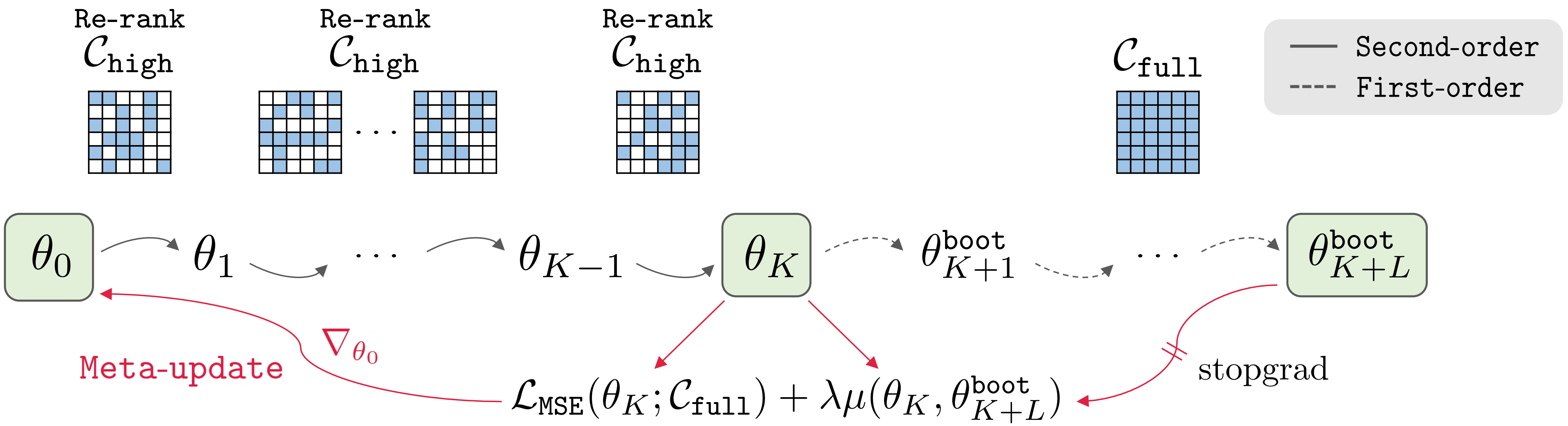
Idea1: Select the **most important context samples** for every adaptation step



High-level overview of GradNCP

Idea1: Select the **most important context samples** for every adaptation step

Idea2: **Correct the error** made by the context pruning

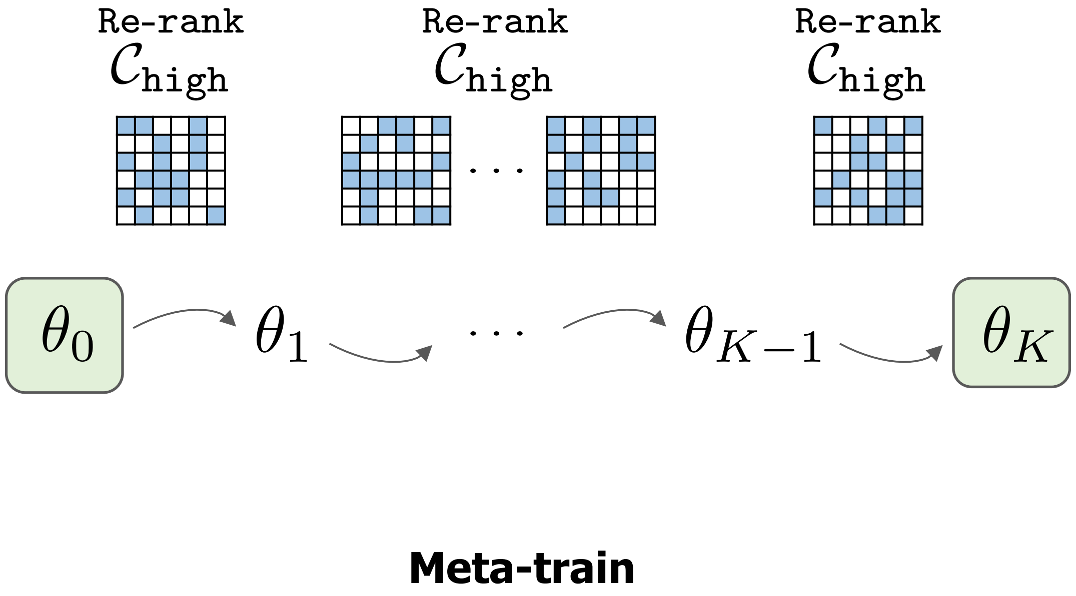


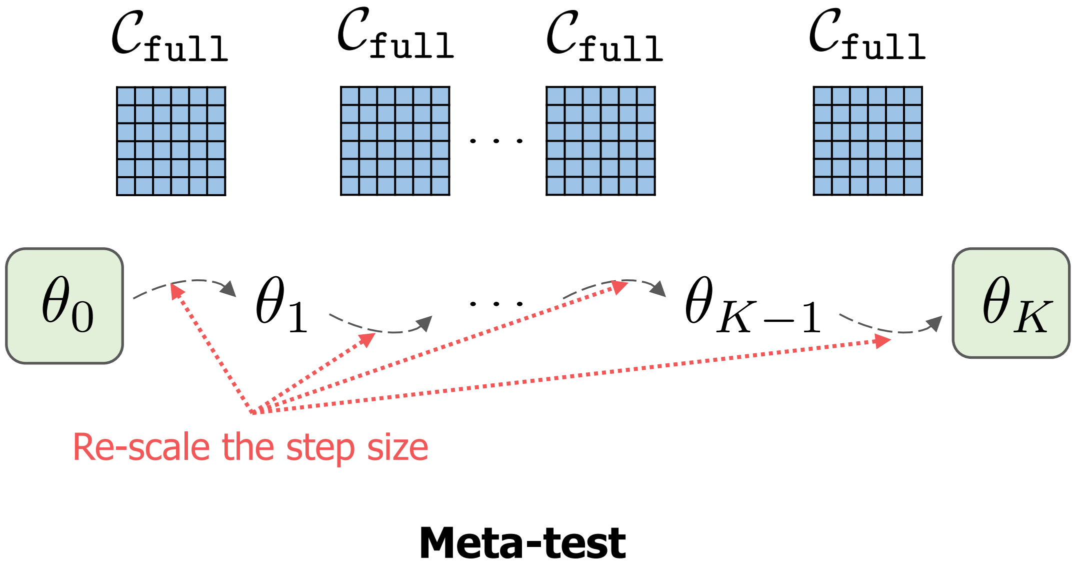
High-level overview of GradNCP

Idea1: Select the **most important context samples** for every adaptation step

Idea2: **Correct the error** made by the context pruning

Idea3: **Re-scale the gradient step size** when using the full context set during meta-testing





Gradient norm-based context pruning

How to select important samples **efficiently?**

- 1. Select a subset of data with the **highest expected immediate improvement in model quality**
- 2. Consider the last layer update only (quite a reasonable choice for NFs [9], also for meta-learning [11])

$$
\begin{aligned}
&\ell\big(\theta_k;\{(\mathbf{x},\mathbf{y})\}\big)-\ell\big(\theta_k';\{(\mathbf{x},\mathbf{y})\}\big) && \text{(Immediate improvement in model quality)}\\
&\approx \ell\big(\theta_k;\{(\mathbf{x},\mathbf{y})\}\big)-\ell\big(\theta_k-\alpha g_k;\{(\mathbf{x},\mathbf{y})\}\big) && \text{(Last layer update only)}\\
&\approx \ell\big(\theta_k;\{(\mathbf{x},\mathbf{y})\}\big)-\Big(\ell\big(\theta_k;\{(\mathbf{x},\mathbf{y})\}\big)-\alpha g_k^\top\nabla_{\theta_k}\ell\big(\theta_k;\{(\mathbf{x},\mathbf{y})\}\big)\Big) && \text{(Taylor approximation)}\\
&=\alpha g_k^\top\nabla_{\theta_k}\ell\big(\theta_k;\{(\mathbf{x},\mathbf{y})\}\big)=\alpha\|g_k\|_2^2 && \text{(Last layer gradient norm)}
\end{aligned}
$$

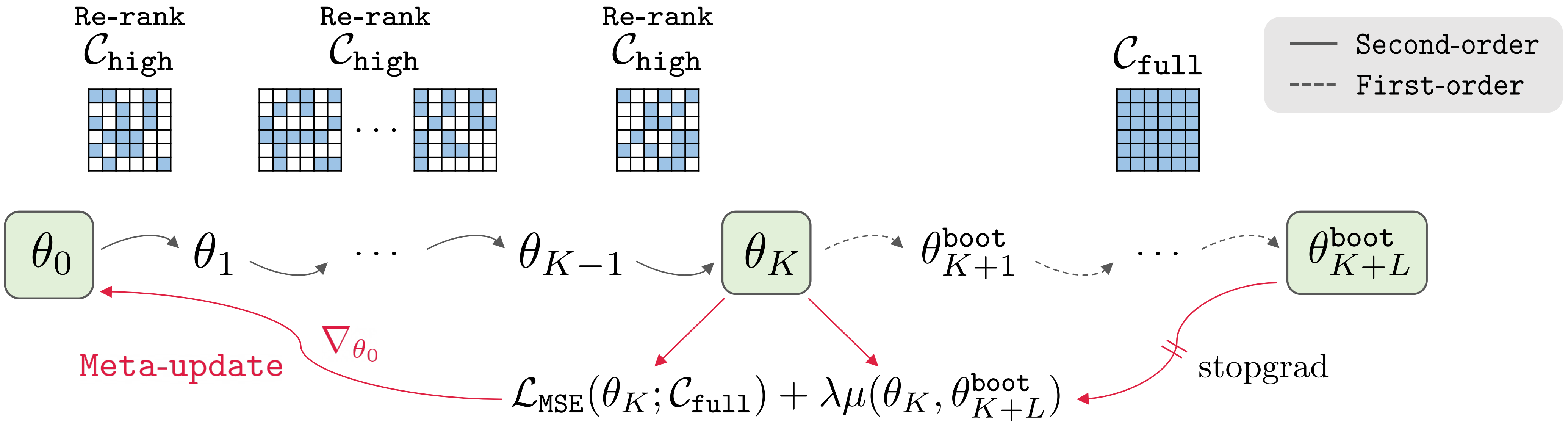
→ This score can be calculated with only a single forward pass

$$R_k^{\mathrm{GradNCP}}(\mathbf{x},\mathbf{y}) := \Big\|\big(\mathbf{y}-f_{\theta_k}(\mathbf{x})\big)\big[\phi_{\theta_k^{\mathrm{base}}}(\mathbf{x}),\mathbf{1}\big]^\top\Big\|$$

$\phi(.)$: penultimate feature
$f(.)$: network output
k: inner adaptation step

[9] Schwarz et al., Modality-Agnostic Variational Compression of Implicit Neural Representations, ICML 2023
[11] Raghu et al., Rapid Learning or Feature Reuse? Towards Understanding the Effectiveness of MAML, ICLR 2020

Bootstrapped correction



Information loss occurs as we prune out some context points

- Idea: further update the network with **the full context set** by **using the first-order gradients**
- After adapting this bootstrapped target $\theta_{K+L}^{\text{boot}}$, minimize the parameter distance between θ_K

This bootstrapped target is also well-known to minimize the myopia (short-horizon bias) of optimization [12]

[12] Flennerhag et al., Bootstrapped Meta-Learning, ICLR 2022

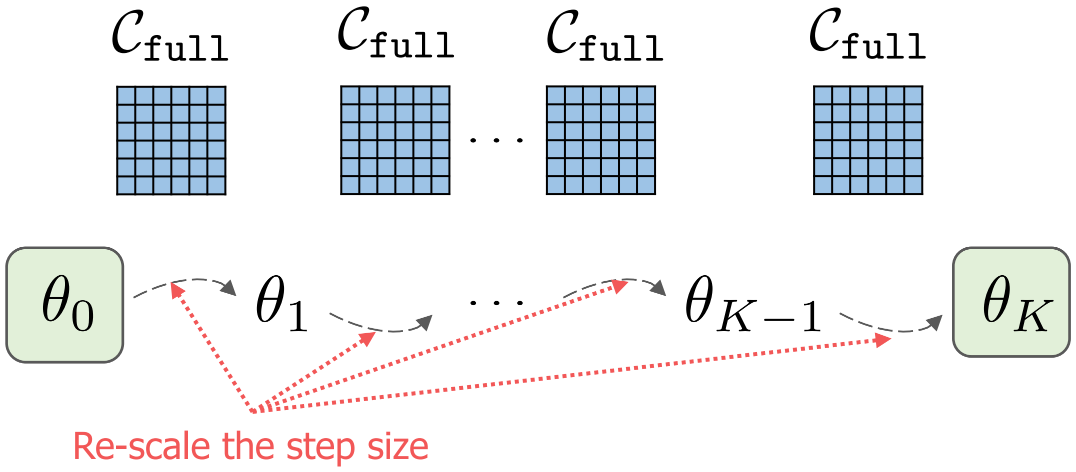
Gradient re-scaling

For meta-testing, we can use first-order gradients for adaptation (second-order is for learning init.)

The **gradient step size deviates a lot** from meta-train (pruned set) and meta-test (full set)

→ Gradient re-scaling: reducing the distributional shift between train/test

- Similar ideas can be found in Dropout (activation scaling) [13]



$$g_t^{\texttt{test}} \leftarrow \underbrace{\boxed{\frac{\|\nabla_{\theta_t}\mathcal{L}_{\texttt{MSE}}(\theta_t;\mathcal{C}_{\texttt{high}})\|}{\|\nabla_{\theta_t}\mathcal{L}_{\texttt{MSE}}(\theta_t;\mathcal{C}_{\texttt{full}})\|}}}_{\text{Scale-factor}} \nabla_{\theta_t}\mathcal{L}_{\texttt{MSE}}(\theta_t;\mathcal{C}_{\texttt{full}})$$

[13] Srivastava et al., Dropout: A Simple Way to Prevent Neural Networks from Overfitting, JMLR 2014

Overall algorithm

Algorithm 1 Meta-training of GradNCP

Input: Initial θ_0, $\{\mathbf{s}_i\}_{i=1}^{N}, \gamma, \alpha, \beta, \lambda, K, L$

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**
      `# Online Context pruning`
      $\mathcal{C}^{k}_{\texttt{high}} = \mathrm{TopK}(\mathcal{C}_{\texttt{full}}; R_k, \gamma)$
      $\theta_{k+1} \leftarrow \theta_k - \alpha \nabla_{\theta_k} \mathcal{L}_{\texttt{MSE}}(\theta_k; \mathcal{C}^{k}_{\texttt{high}})$
    end for
    `# Generate target in L steps`
    $\theta^{\texttt{boot}}_{K+1} \leftarrow \theta_K - \alpha \nabla_{\theta_K} \mathcal{L}_{\texttt{MSE}}(\theta_K; \mathcal{C}_{\texttt{full}})$
    $\dots$
    $\mathcal{L}^{b}_{\texttt{total}} = \mathcal{L}_{\texttt{MSE}}(\theta_K; \mathcal{C}_{\texttt{full}}) + \lambda \mu(\theta_K, \theta^{\texttt{boot}}_{K+L})$
  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

Output: θ_0

Algorithm 2 Meta-testing of GradNCP

Input: Test signal $\mathbf{s}$, learned initialization θ_0, K_{test}

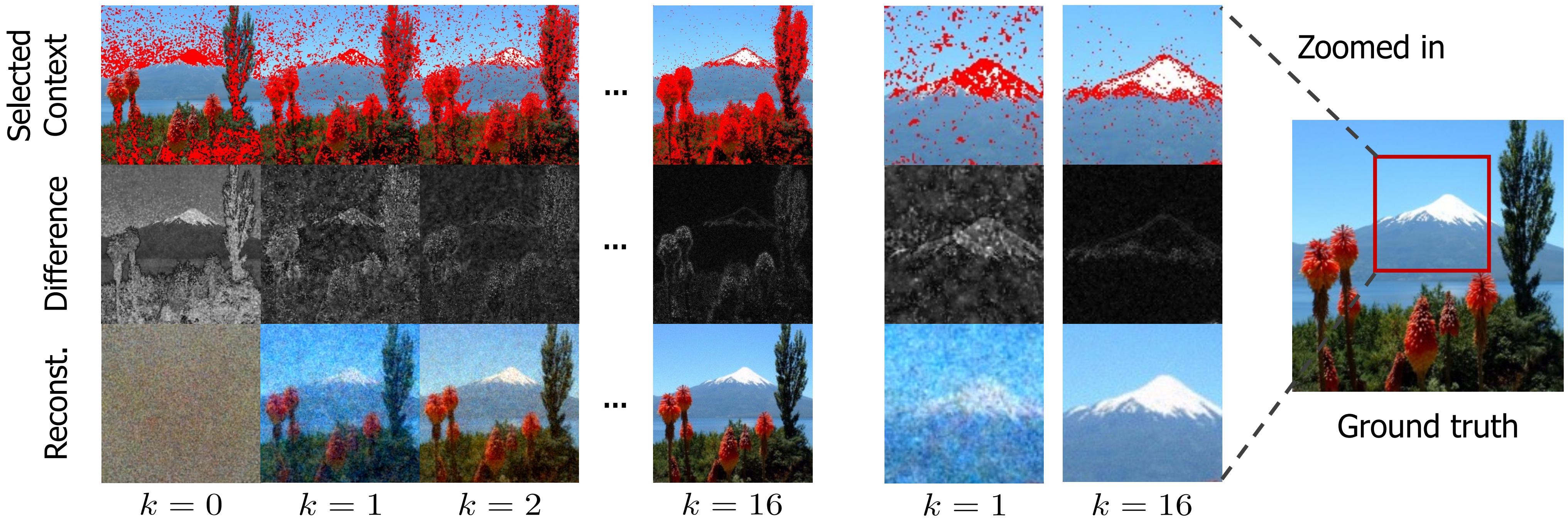
Extract context $\mathcal{C}_{\texttt{full}}$ from $\mathbf{s}$.
`# Where typically` $K_{\texttt{test}} > K + L$
for all $t = 0$ to $K_{\text{test}} - 1$ **do**
  `# Context pruning`
  $\mathcal{C}_{\texttt{high}} = \mathrm{TopK}(\mathcal{C}_{\texttt{full}}; R_t, \gamma)$
  `# Compute gradient scaling`
  $g_t^{\texttt{test}} = \frac{\|\nabla_{\theta_t} \mathcal{L}_{\texttt{MSE}}(\theta_t, \mathcal{C}_{\texttt{high}})\|}{\|\nabla_{\theta_t} \mathcal{L}_{\texttt{MSE}}(\theta_t, \mathcal{C}_{\texttt{full}})\|} \nabla_{\theta_t} \mathcal{L}_{\texttt{MSE}}(\theta_t, \mathcal{C}_{\texttt{full}})$
  `# Adaptation with full context`
  $\theta_{t+1} \leftarrow \theta_t - \alpha \cdot g_t^{\texttt{test}}$
end for

Output: $\mathcal{L}_{\texttt{MSE}}(\theta_T, \mathcal{C}_{\texttt{full}}), \theta_T$

Visualization

Visualization of the selected context point

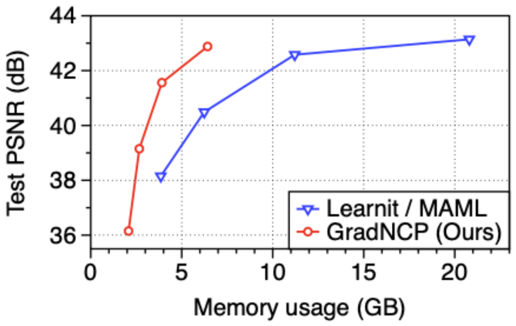
- Interestingly, it automatically **captures the global shape first**, then captures the high-frequency details
- → prior works do this in a hand-craft manner [14]



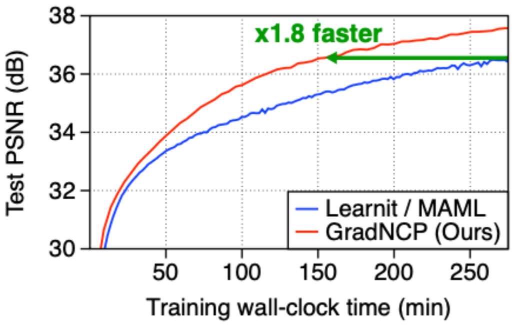
[14] Landgraf et al., Pins: Progressive implicit networks for multi-scale neural representations, ICML 2022

Results

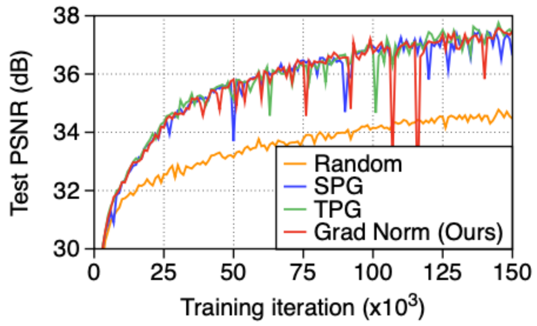
Efficiency of GradNCP



(a) Memory consumption

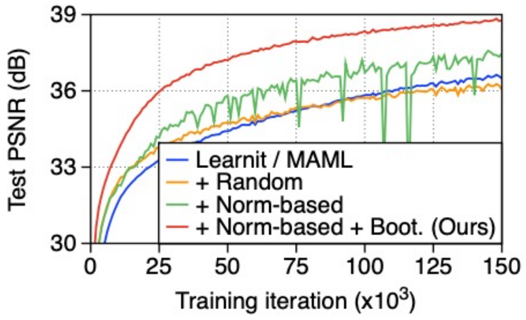


(b) Runtime efficiency

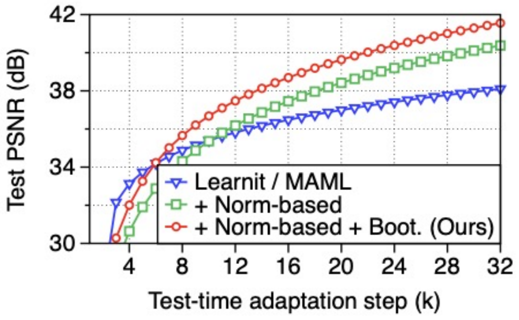


(c) Context scoring quality

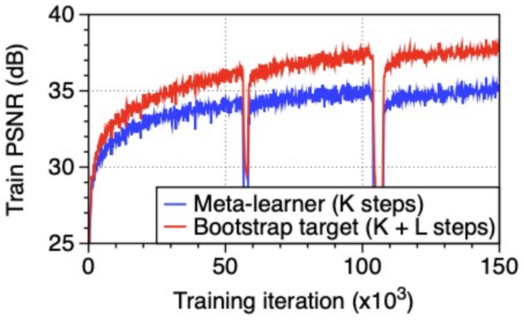
Effectiveness of Bootstrapped correction



(a) GradNCP component analysis



(b) Myopia



(c) Bootstrap target performance

Results: Quantitative

GradNCP achieves state-of-the-art performance on meta-learning neural fields in all modalities

Image

	CelebA (178 × 178)	**Imagenette** (178 × 178)	**Text** (178 × 178)
Random Init.	19.94 / 0.532 / 0.708	18.57 / 0.443 / 0.810	15.37 / 0.574 / 0.755
TransINR [6]	32.37 / 0.913 / 0.068	28.58 / 0.850 / 0.165	22.70 / 0.898 / 0.085
IPC [29]	35.93 / - / -	38.46 / - / -	- / - / -
Learnit / MAML [61]	38.28 / 0.964 / 0.010	35.66 / 0.950 / 0.014	30.31 / 0.956 / 0.018
GradNCP (Ours)	**40.60 / 0.976 / 0.005**	**38.72 / 0.972 / 0.005**	**32.33 / 0.976 / 0.007**
	ImageNet (256 × 256)	**AFHQ** (512 × 512)	**CelebA-HQ** (1024 × 1024)
Random Init.	18.72 / 0.434 / 0.839	18.57 / 0.488 / 0.856	12.21 / 0.574 / 0.820
TransINR [6]	28.01 / 0.818 / 0.199	23.43 / 0.592 / 0.573	——— OOM ———
Learnit / MAML [61]	31.44 / 0.887 / 0.100	28.58 / 0.751 / 0.354	27.66 / 0.781 / 0.513
GradNCP (Ours)	**32.52 / 0.898 / 0.068**	**29.61 / 0.786 / 0.286**	**28.90 / 0.789 / 0.438**

Audio

Method	PSNR (↑) 1 sec	PSNR (↑) 3 sec
TransINR [6]	39.22	33.17
IPC [29]	40.11	35.38
Learnit / MAML [61]	39.55	31.39
GradNCP (Ours)	**43.25**	**36.24**

Manifold

Method	PSNR (↑)
Learnit / MAML [61]	64.91
GradNCP (Ours)	**75.11**

Video

Resolution	Network	Method	PSNR (↑)	SSIM (↑)	LPIPS (↓)
128×128×16	SIREN	TransINR [6]	15.14	0.360	0.636
		Learnit / MAML [61]	25.46	0.720	0.363
		GradNCP (Ours)	**26.92**	**0.781**	**0.223**
	NeRV	Learnit (MAML) [61]	28.86	0.871	0.140
		GradNCP (Ours)	**35.28**	**0.959**	**0.015**
256×256×32	SIREN	TransINR [6]	———	OOM	———
		Learnit / MAML [61]	———	OOM	———
		GradNCP (Ours)	**22.92**	**0.640**	**0.521**
	NeRV	Learnit / MAML [61]	23.75	0.659	0.422
		GradNCP (Ours)	**28.65**	**0.842**	**0.201**

NeRFs

Method	PSNR
Learnit [61]	22.80
TransINR [6]	23.78
GradNCP (Ours)	**24.06**

General (Few-shot)

Method	5-way 100-shot	10-way 50-shot
MAML [15]	66.03±0.82	48.95±0.52
GradNCP (Ours)	**73.45±0.83**	**55.71±0.49**

Results: Qualitative

GradNCP achieves state-of-the-art performance on meta-learning neural fields in all modalities

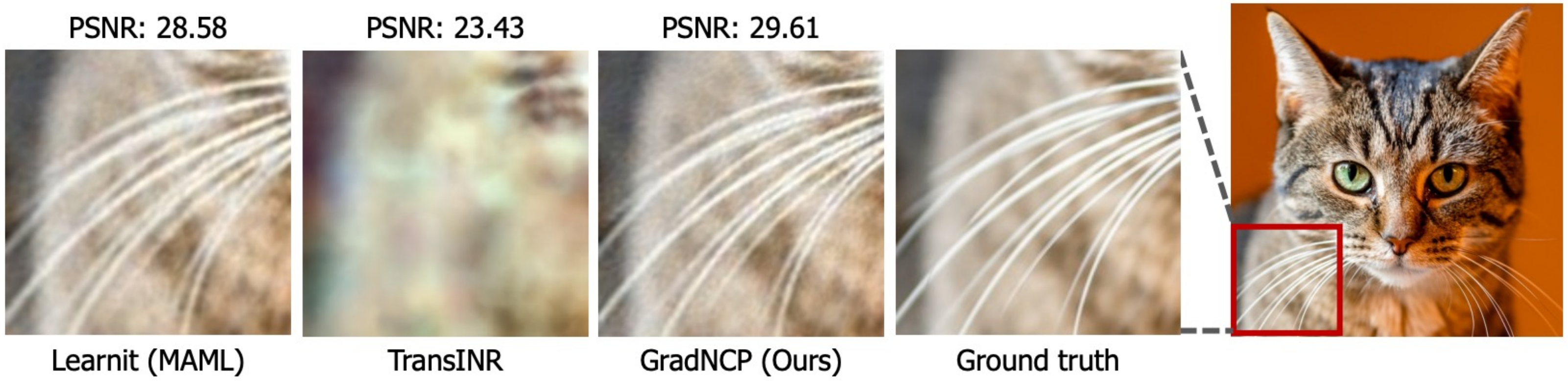
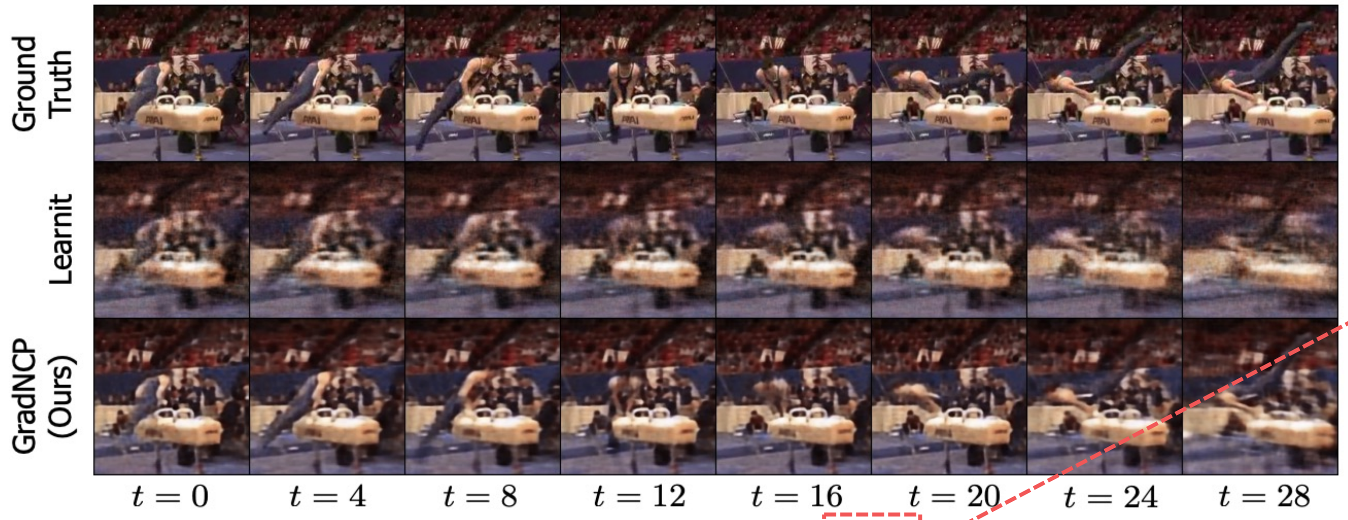


Figure 6: Qualitative comparison between GradNCP and baselines on AFHQ (512×512).



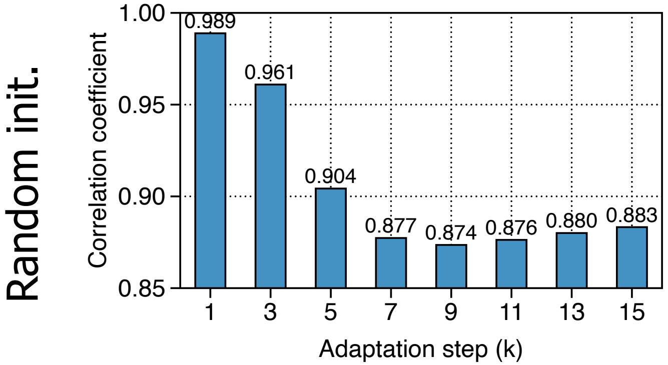
(b) UCF-101 (256×256×32) on NeRV

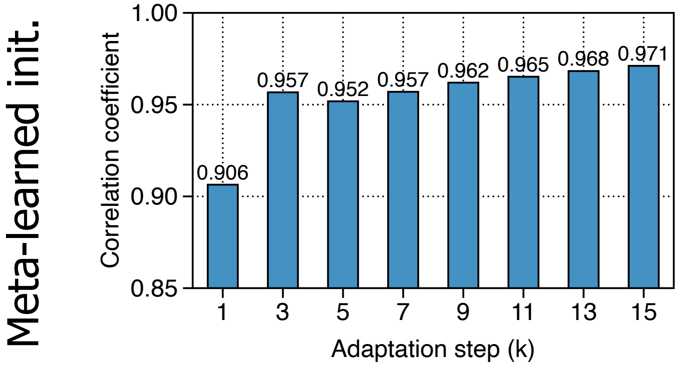
+ GradNCP is **model-agnostic framework** i.e., can be used for any neural fields

Analysis

Quality of the last layer approximation? Shows a high correlation with the full-layer gradient norm

- \+ **Meta-learning automatically learns to improve this correlation**





Necessary to use gradient re-scaling when using the full context set during meta-testing

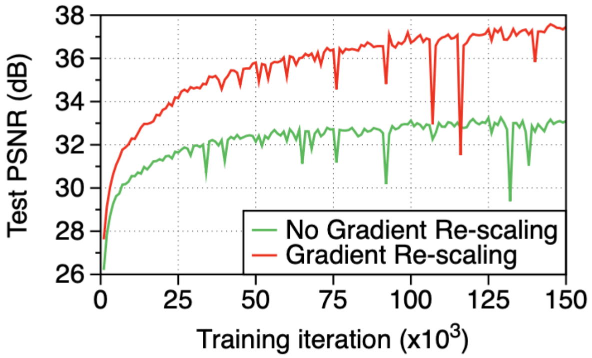
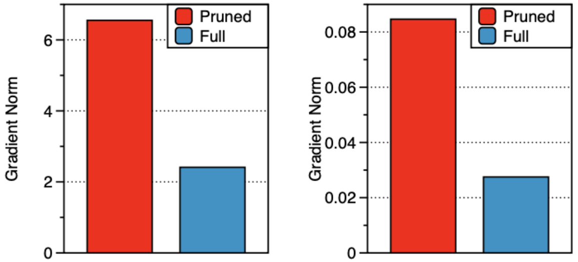


Figure 7: Effect of meta-test time gradient re-scaling. We apply re-scaling when adapting with full context set on SIREN trained with GradNCP.



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

Figure 8: Gradient norm of the full context set $\mathcal{C}_{\texttt{full}}$ and the gradient norm-based pruned context set $\mathcal{C}^k_{\texttt{high}}$ at iteration k.

Reason for using gradient re-scaling:
the gradient step size differs when using full/pruned context

Thank you for your attention ☺

For any more questions, please send us an email!

Email: jihoontack@gmail.com