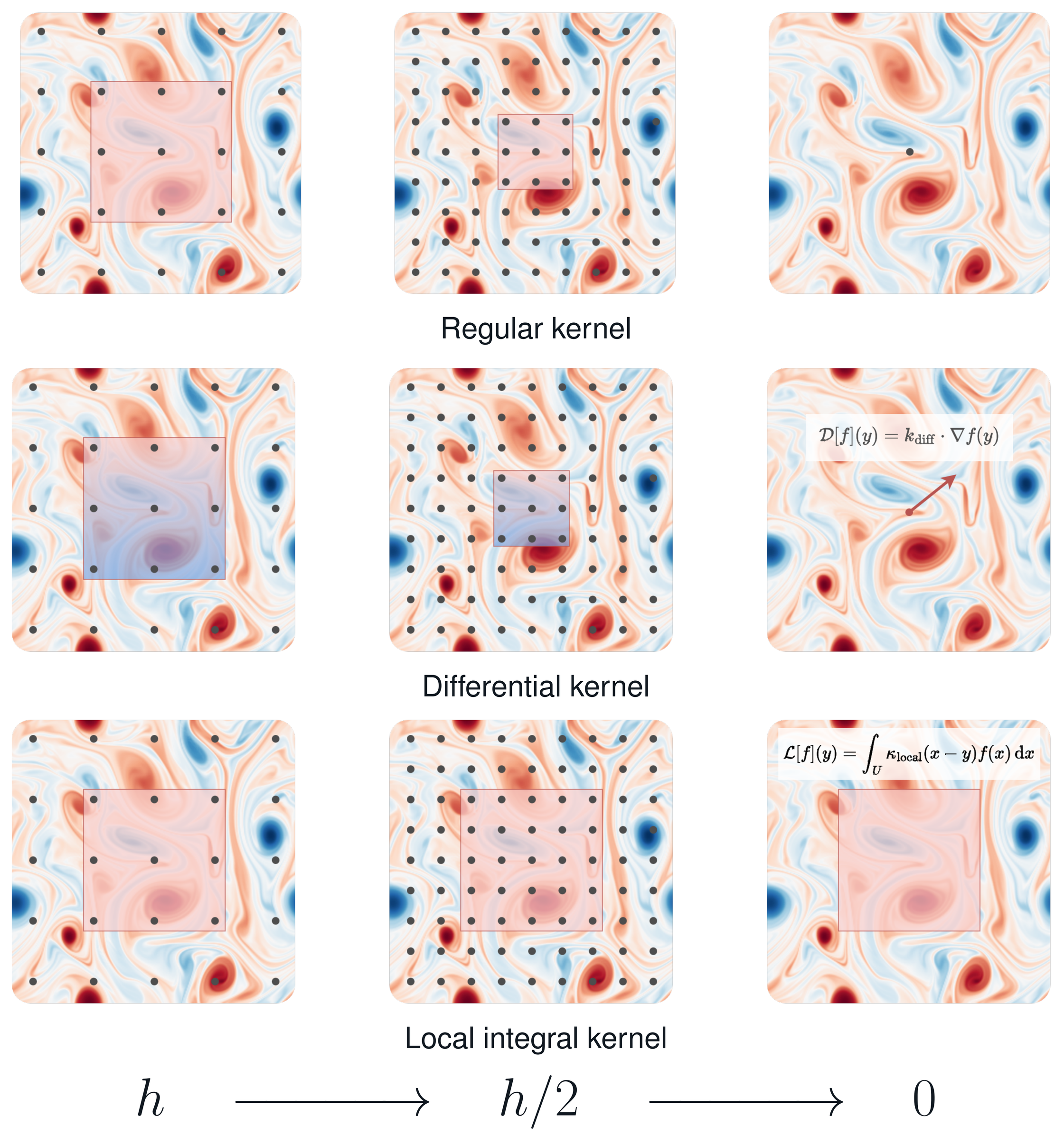
Neural Operators with Localized Integral and Differential Kernels

Miguel Liu-Schiaffini[*1], **Julius Berner**[*1], **Boris Bonev**[*2], **Thorsten Kurth**[2], **Kamyar Azizzadenesheli**[2], **Anima Anandkumar**[1]

[1]Caltech, [2]NVIDIA, [*]equal contribution

Motivation

Learning **local operators** is necessary for a variety of applications: turbulent fluid flows, hyperbolic PDEs, etc.



In neural networks, local operations have typically been performed by standard **convolutional layers**. However, regular kernels converge to pointwise operators when the grid width h is refined. They are **not discretization-agnostic**.

Architecture	Efficient	Receptive field	no input downsampling
GNO	✗	local/global	✓
FNO	✓	global	✓
CNO / U-Net	✓	local	✗
FNO + integral	✓	local/global	✓
FNO + differential	✓	local/global	✓
SFNO	✓	global	✓
SFNO + integral	✓	local/global	✓

Existing neural operator methods are either global, inefficient, or require interpolation/downsampling of the input.

In this paper, we develop a framework for introducing **local and differential operators** into neural operator architectures.

Differential layers

We introduce **differential layers**, which provably converge to differential operators as resolution increases.

We show that we can learn different directional derivatives with two minor modifications of standard convolutional kernels:

- Subtracting the mean of the kernel,
- Scaling the kernel by the inverse grid width $\frac{1}{h}$.

Integral kernel layers

We also use **discrete-continuous convolutions** to define discretization-agnostic local convolutions in physical space.

Consider the group convolution on the group G:

$$\mathrm{GroupConv}_\kappa[v](g) = (\kappa \star v)(g) = \int_G \kappa(g^{-1}x)\cdot v(x)\,\mathrm{d}\mu(x),$$

with group actions $g, x \in G$ and $\mathrm{d}\mu(x)$ the invariant Haar measure.

To formulate a neural operator, we use the framework of discrete-continuous convolutions (Ocampo et al., 2022), evaluating the group action analytically and the integral discretely:

$$(k \star v)(g_i) \approx \sum_{j=1}^{m} \kappa(g_i^{-1}x_j)\cdot v(x_j)\,q_j,$$

where q_j are quadrature weights associated with points x_j and $K_{ij} = \kappa(g_i^{-1}x_j)$ for output position g_i.

To achieve discretization convergence, we parameterize the convolutional kernel κ as

$$\kappa = \sum_{\ell=1}^{L} \theta^{(\ell)}\kappa^{(\ell)},$$

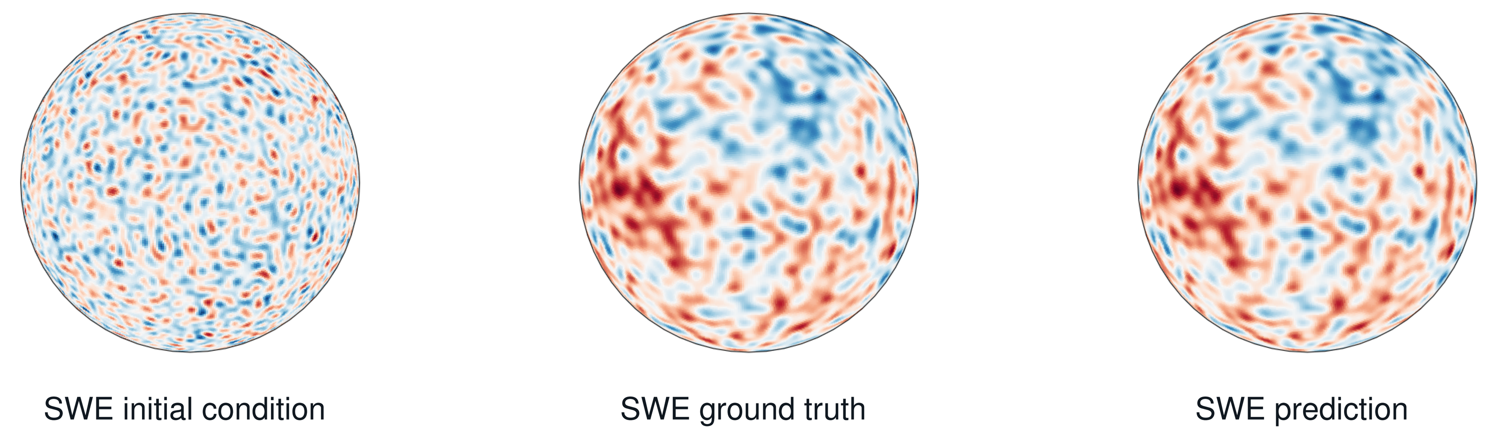
where $\kappa^{(\ell)}$ are predefined (continuous) basis functions, and $\theta^{(\ell)}$ are learnable parameters.

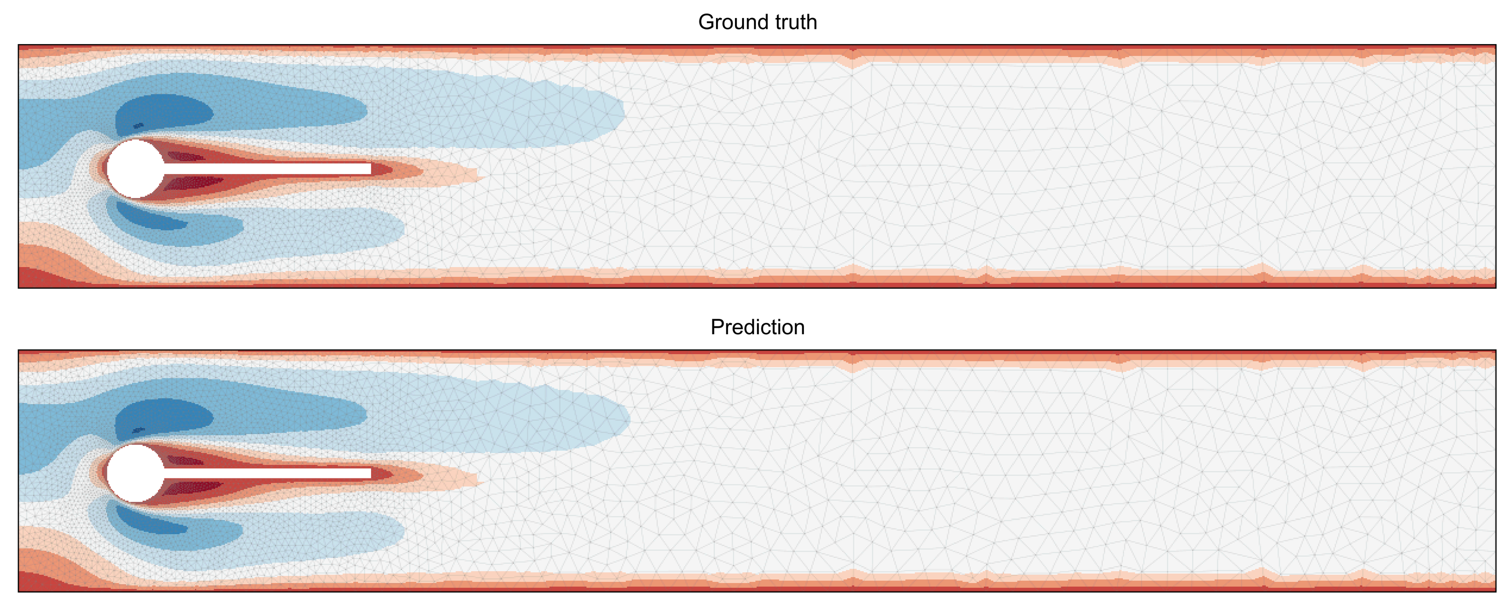
This formulation allows for **local convolutions on manifolds** (e.g., sphere $\mathbb{S}^2$) and on **irregular grids** with appropriate quadrature weights.

Experiments

We evaluate our local layers on several problems:

- Darcy flow
- 2D Navier-Stokes (NS) equations
- Spherical shallow water equations (SWE)
- 2D reaction-diffusion equation
- Flow past a cylinder (irregular grid)





Horizontal velocity on flow over a cylinder

Results on Navier-Stokes (Re 5000):

Model	Relative L^2-Error	
	1 step	5 steps
U-Net	$1.674\cdot10^{-1}$	$5.115\cdot10^{-1}$
FNO	$1.381\cdot10^{-1}$	$2.360\cdot10^{-1}$
FNO + diff. kernel (ours)	$1.073\cdot10^{-1}$	$2.129\cdot10^{-1}$
FNO + local integral kernel (ours)	$1.110\cdot10^{-1}$	$2.183\cdot10^{-1}$
FNO + local int. + diff. kernel (ours)	$\mathbf{9.022\cdot10^{-2}}$	$\mathbf{1.956\cdot10^{-1}}$

In our experiments, we outperform baselines by up to 72% on relative L^2 error. We find that **fusion of global and local** operators **outperforms** models with **purely local or global** operators.

Paper: arxiv.org/abs/2402.16845
Code (soon): github.com/neuraloperator/neuraloperator