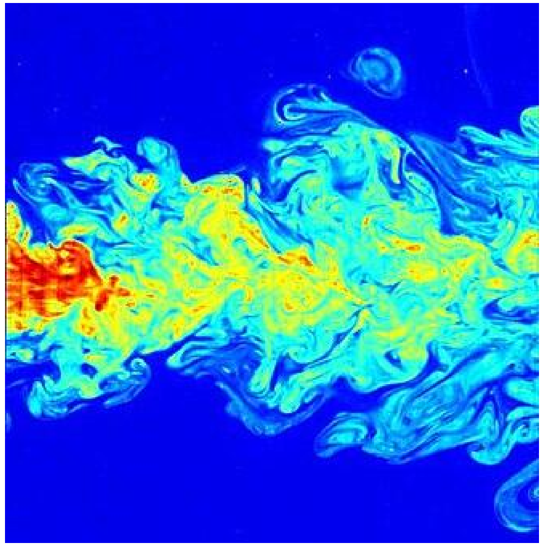
ICML 2024
International Conference On Machine Learning
Forty-first International Conference on Machine Learning

HelmFluid: Learning Helmholtz Dynamics for Interpretable Fluid Prediction

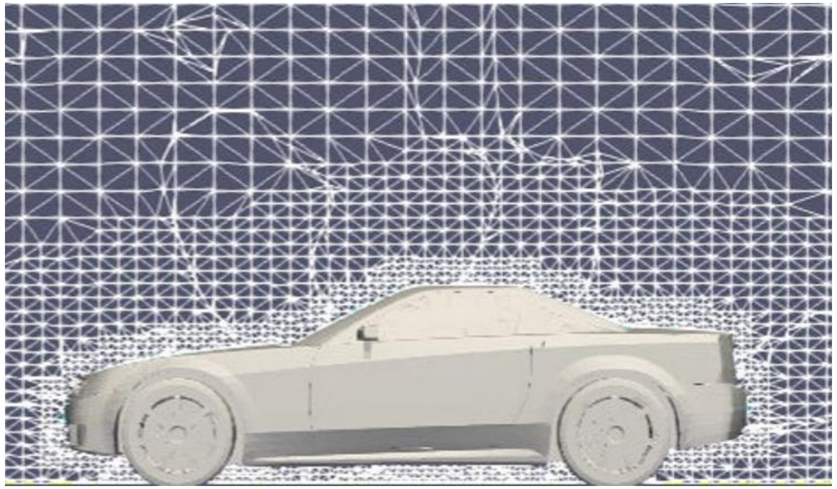
Lanxiang Xing *[1] Haixu Wu *[1] Yuezhou Ma [1] Jianmin Wang [1] Mingsheng Long [1]



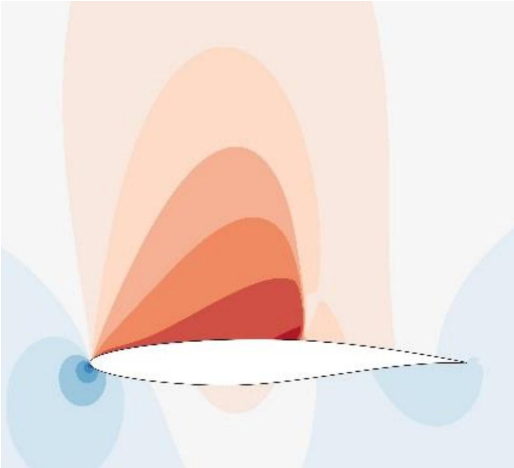

Fluid Prediction



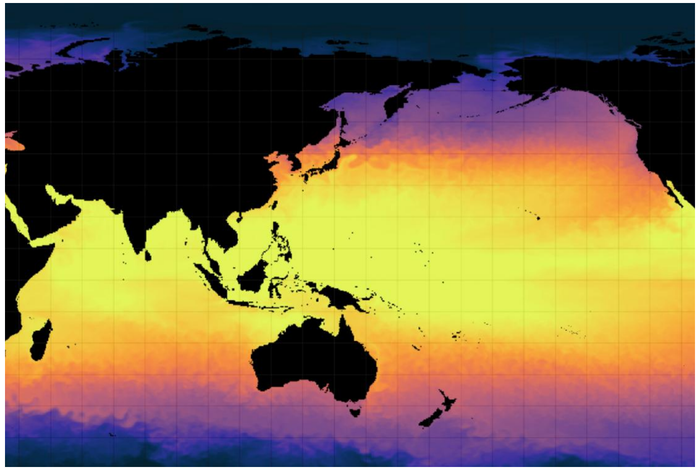
Navier-Stokes Equation: $$\rho \frac{\mathbf{Dv}}{\mathbf{D}t} = \nabla \cdot \mathbb{P} + \rho \mathbf{f}$$



(a) Vehicle shape design



(b) Airfoil Design



(c) Ocean Current Prediction

Computational Fluid Dynamics

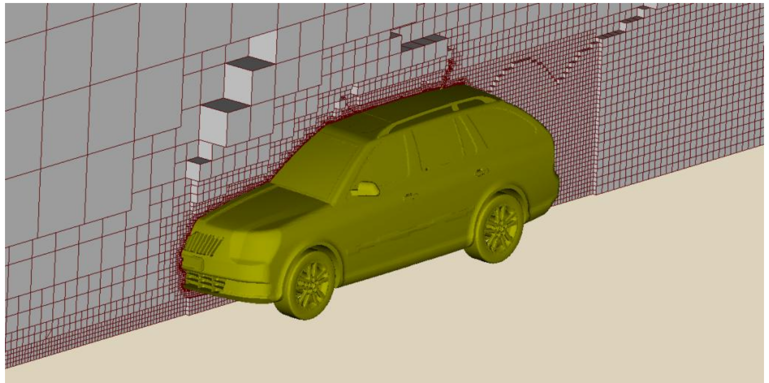
Definition of geometry & physical bounds

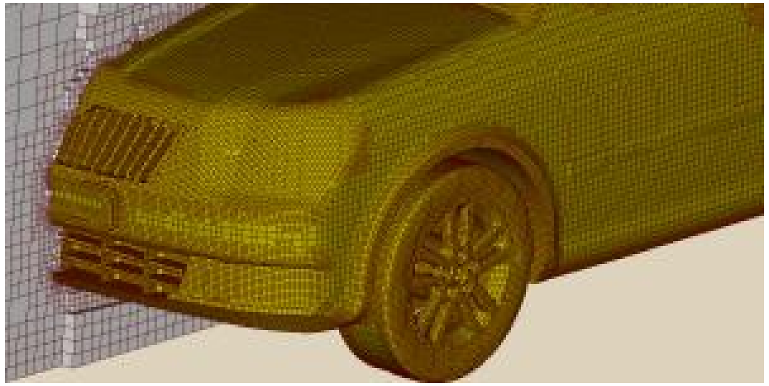
Discretization into mesh

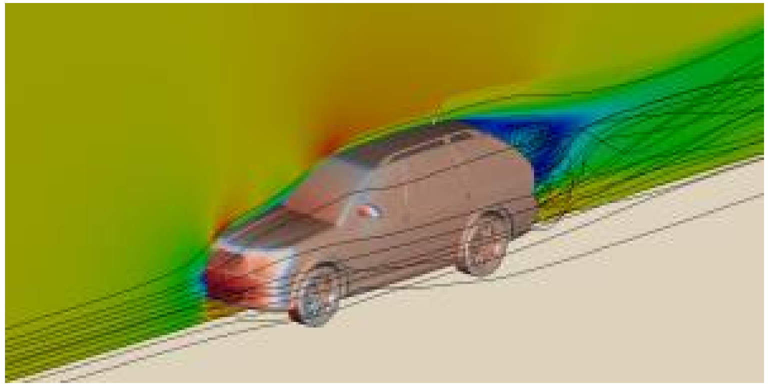
Physical Modeling

Iterative equations solving

Postprocessing Analysis







https://cfd.direct/openfoam/computational-fluid-dynamics/

Computational Fluid Dynamics

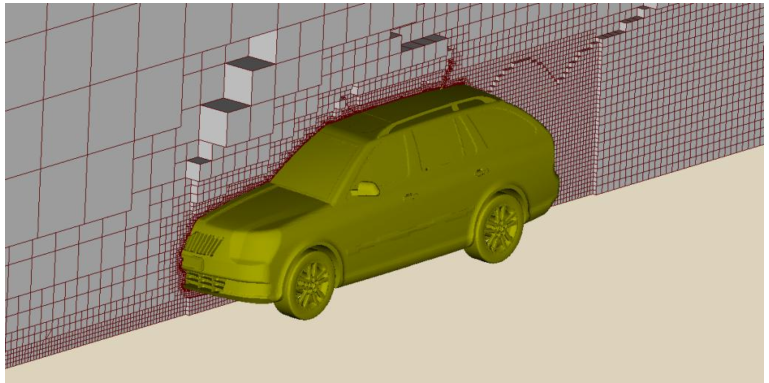
Definition of geometry & physical bounds

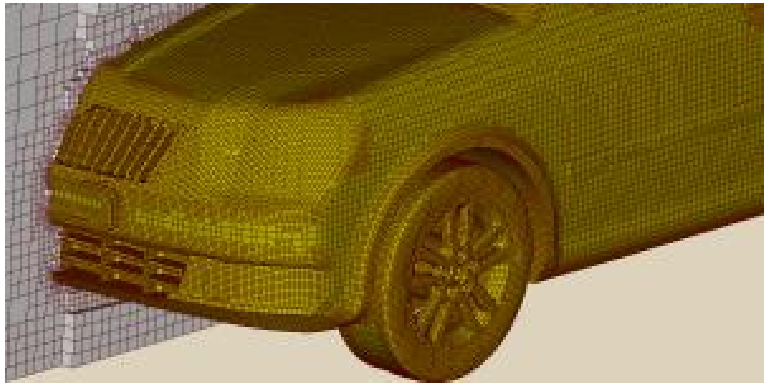
Discretization into mesh

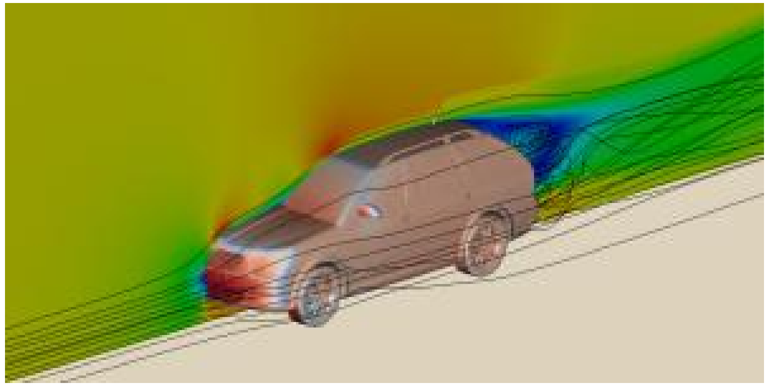
Physical Modeling

Iterative equations solving

Postprocessing Analysis







Slow! **Partial Observation?**

https://cfd.direct/openfoam/computational-fluid-dynamics/

Computational Fluid Dynamics

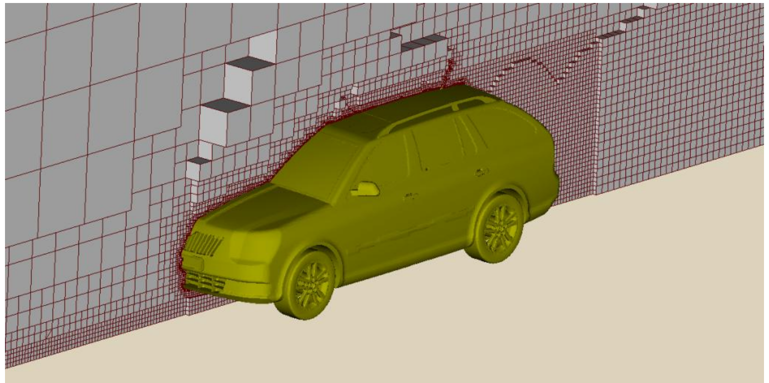
Definition of geometry & physical bounds

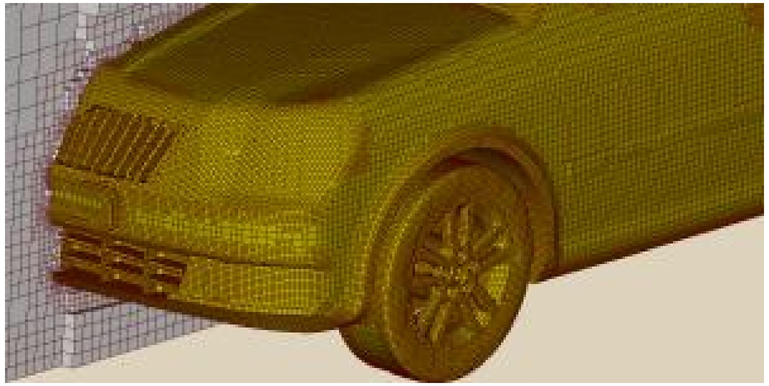
Discretization into mesh

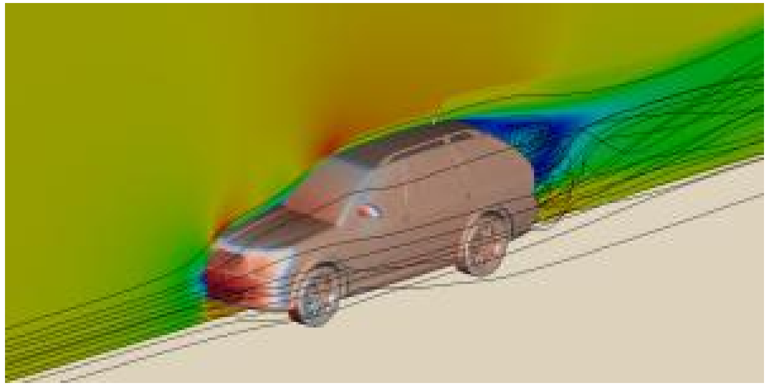
Physical Modeling

Iterative equations solving

Postprocessing Analysis







Slow! **Partial Observation?**

Data-driven deep models for fluid prediction ✓

https://cfd.direct/openfoam/computational-fluid-dynamics/

Neural Fluid Simulator

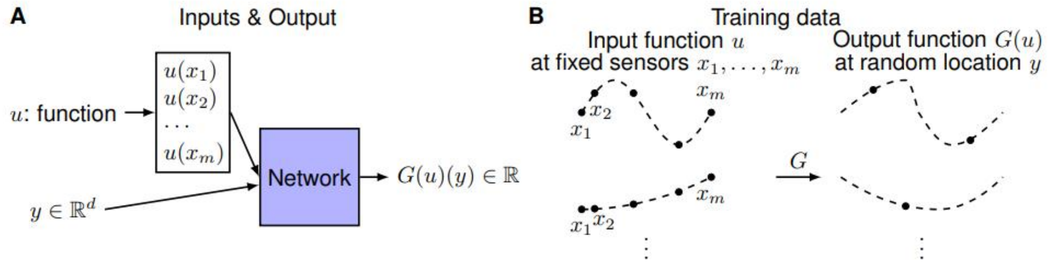
Universal Approximation Theorem for Operator:

The potential application of neural networks to learn nonlinear operators from data

Theorem 1 (Universal Approximation Theorem for Operator). *Suppose that σ is a continuous non-polynomial function, X is a Banach Space, $K_1 \subset X$, $K_2 \subset \mathbb{R}^d$ are two compact sets in X and $\mathbb{R}^d$, respectively, V is a compact set in $C(K_1)$, G is a nonlinear continuous operator, which maps V into $C(K_2)$. Then for any $\epsilon > 0$, there are positive integers n, p, m, constants $c_i^k, \xi_{ij}^k, \theta_i^k, \zeta_k \in \mathbb{R}$, $w_k \in \mathbb{R}^d$, $x_j \in K_1$, $i = 1, \ldots, n$, $k = 1, \ldots, p$, $j = 1, \ldots, m$, such that*

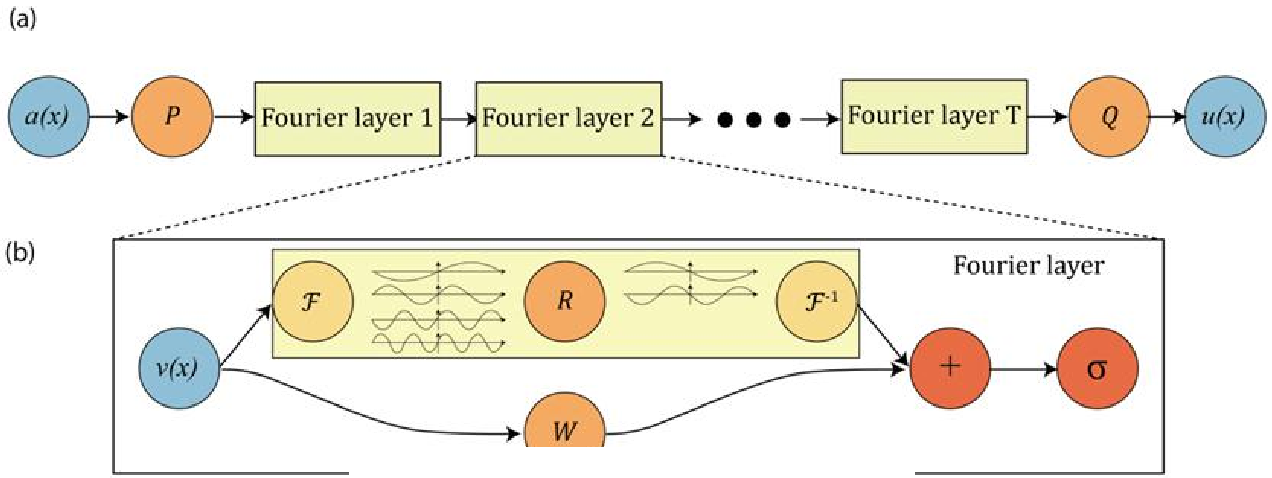
$$\left| G(u)(y) - \sum_{k=1}^{p} \underbrace{\sum_{i=1}^{n} c_i^k \sigma \left(\sum_{j=1}^{m} \xi_{ij}^k u(x_j) + \theta_i^k \right)}_{branch} \underbrace{\sigma(w_k \cdot y + \zeta_k)}_{trunk} \right| < \epsilon \quad (1)$$

holds for all $u \in V$ and $y \in K_2$.

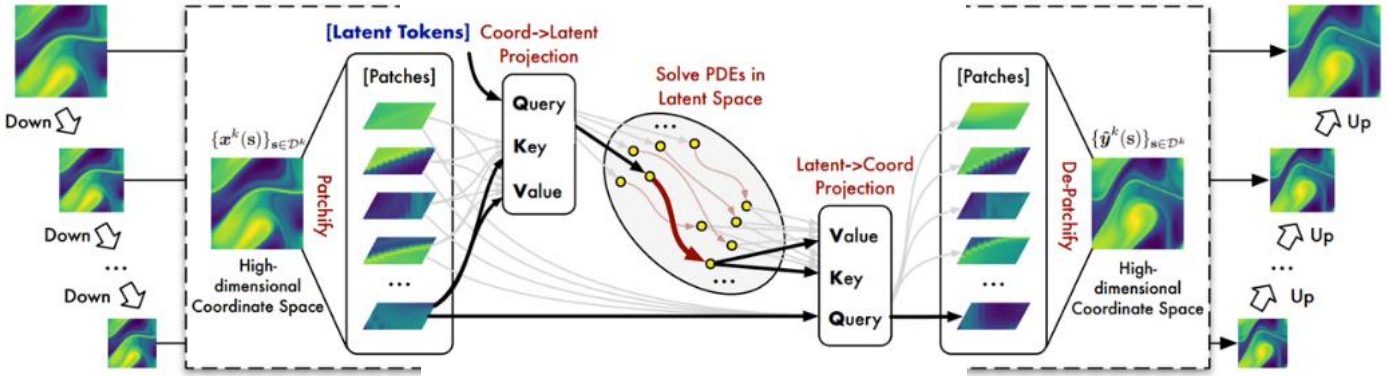


Lu, et al. Deeponet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. JMM 2020

Neural Fluid Simulator



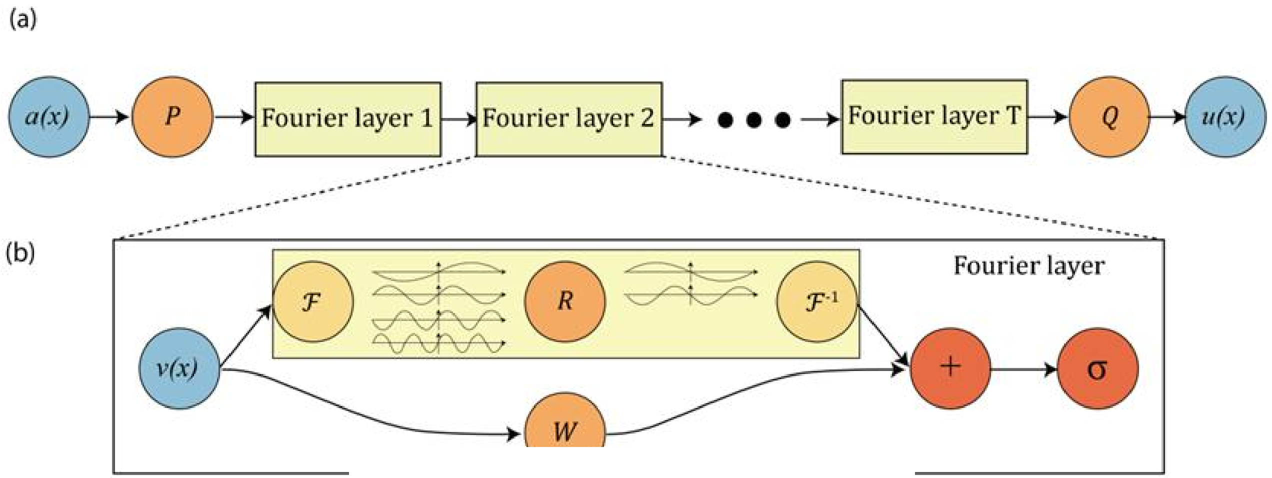
Fourier Neural Operator



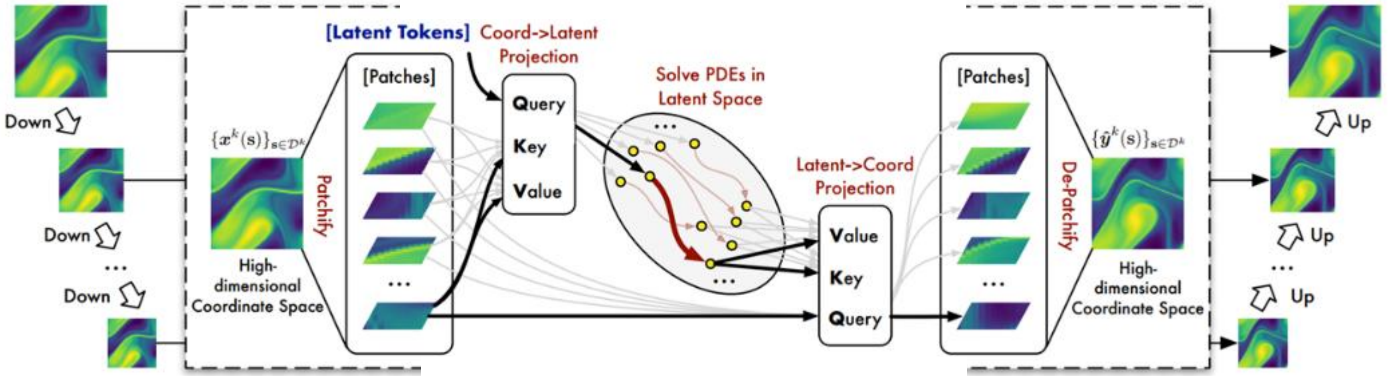
Latent Spectral Models

Li, et al. Fourier neural operator for parametric partial differential equation. ICLR 2021
Wu, et al. Solving high-dimensional pdes with latent spectral models. ICML 2023

Neural Fluid Simulator



Fourier Neural Operator



Latent Spectral Models

No interpretable evidence

Li, et al. Fourier neural operator for parametric partial differential equation. ICLR 2021
Wu, et al. Solving high-dimensional pdes with latent spectral models. ICML 2023

Neural Fluid Simulator

Physics informed neural networks:
Equation as loss function

$$\nabla \cdot \vec{v} = 0 \qquad \text{incompressibility on } \Omega \qquad (1)$$

$$\rho\dot{\vec{v}} = \rho\left(\frac{\partial \vec{v}}{\partial t} + (\vec{v}\cdot\nabla)\,\vec{v}\right) = -\nabla p + \mu\Delta\vec{v} + \vec{f} \qquad \text{conservation of momentum on } \Omega \qquad (2)$$

$$\vec{v} = \vec{v}_d \qquad \text{Dirichlet boundary condition on } \partial\Omega \qquad (3)$$

Wande, et al. Learning Incompressible Fluid Dynamics from Scratch--Towards Fast, Differentiable Fluid Models that Generalize. ICLR 2021
Raiss, et al. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational physics 2019

Neural Fluid Simulator

Physics informed neural networks:

Equation as loss function

$$\nabla \cdot \vec{v} = 0 \qquad \text{incompressibility on } \Omega \qquad (1)$$

$$\rho \dot{\vec{v}} = \rho \left(\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \nabla)\, \vec{v} \right) = -\nabla p + \mu \Delta \vec{v} + \vec{f} \qquad \text{conservation of momentum on } \Omega \qquad (2)$$

$$\vec{v} = \vec{v}_d \qquad \text{Dirichlet boundary condition on } \partial\Omega \qquad (3)$$

$$\Downarrow$$

$$L_d = \| \nabla \cdot \vec{v} \|^2 \qquad \text{divergence loss on } \Omega \qquad (8)$$

$$L_p = \left\| \rho \left(\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \nabla)\, \vec{v} \right) + \nabla p - \mu \Delta \vec{v} - \vec{f} \right\|^2 \qquad \text{momentum loss on } \Omega \qquad (9)$$

$$L_b = \| \vec{v} - \vec{v}_d \|^2 \qquad \text{boundary loss on } \partial\Omega \qquad (10)$$

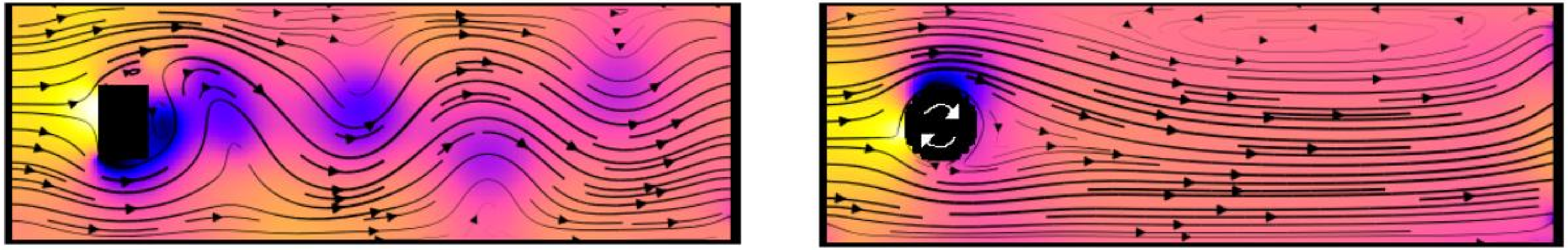
Wande, et al. Learning Incompressible Fluid Dynamics from Scratch--Towards Fast, Differentiable Fluid Models that Generalize. ICLR 2021

Neural Fluid Simulator

$$L_d = \|\nabla \cdot \vec{v}\|^2 \qquad \text{divergence loss on } \Omega \qquad (8)$$

$$L_p = \left\|\rho\left(\frac{\partial \vec{v}}{\partial t} + (\vec{v}\cdot\nabla)\,\vec{v}\right) + \nabla p - \mu\Delta\vec{v} - \vec{f}\right\|^2 \qquad \text{momentum loss on } \Omega \qquad (9)$$

$$L_b = \|\vec{v} - \vec{v}_d\|^2 \qquad \text{boundary loss on } \partial\Omega \qquad (10)$$



Learning Kármán vortex street & Magnus effect **from scratch**

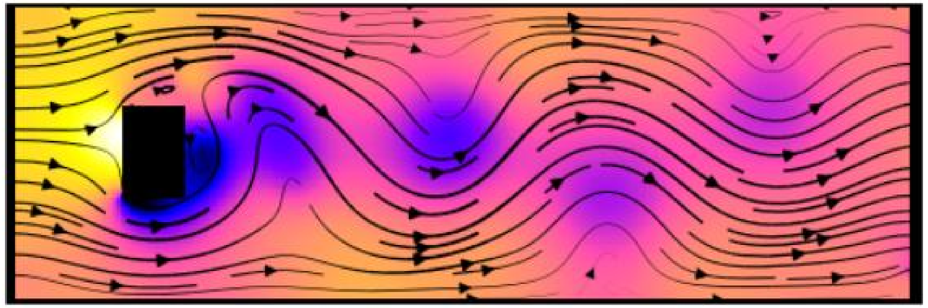
Wande, et al. Learning Incompressible Fluid Dynamics from Scratch--Towards Fast, Differentiable Fluid Models that Generalize. ICLR 2021

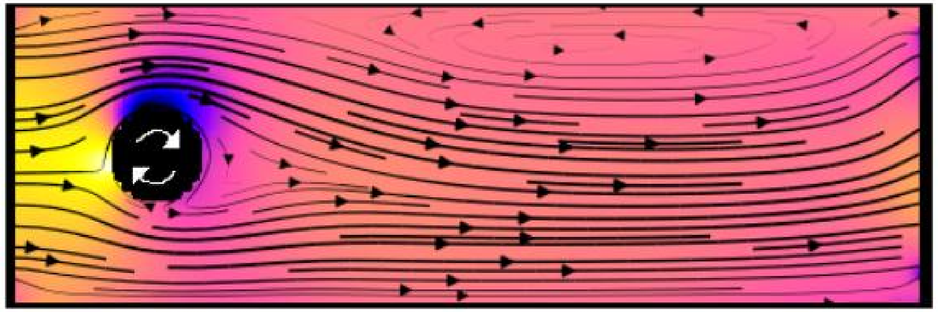
Neural Fluid Simulator

$$L_d = \|\nabla \cdot \vec{v}\|^2 \qquad \text{divergence loss on } \Omega \qquad (8)$$

$$L_p = \left\| \rho \left(\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \nabla)\, \vec{v} \right) + \nabla p - \mu \Delta \vec{v} - \vec{f} \right\|^2 \qquad \text{momentum loss on } \Omega \qquad (9)$$

$$L_b = \|\vec{v} - \vec{v}_d\|^2 \qquad \text{boundary loss on } \partial\Omega \qquad (10)$$





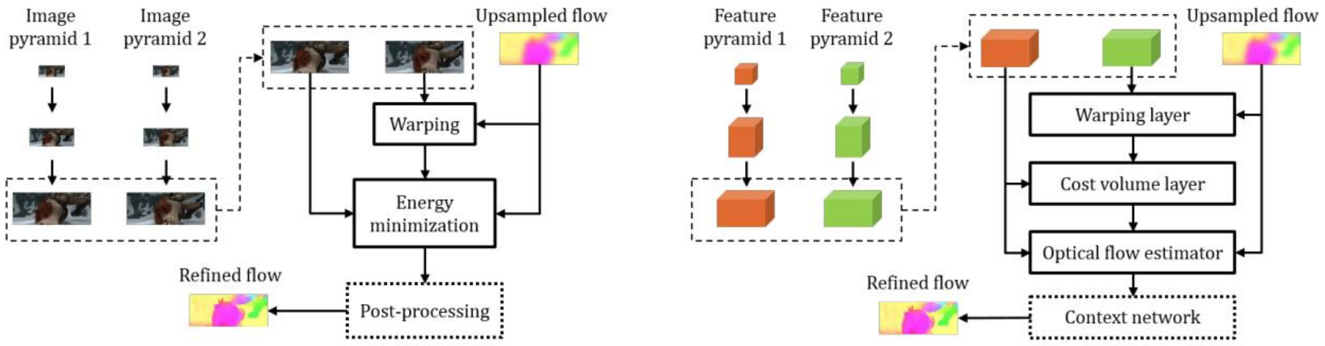
Learning Kármán vortex street & Magnus effect **from scratch**

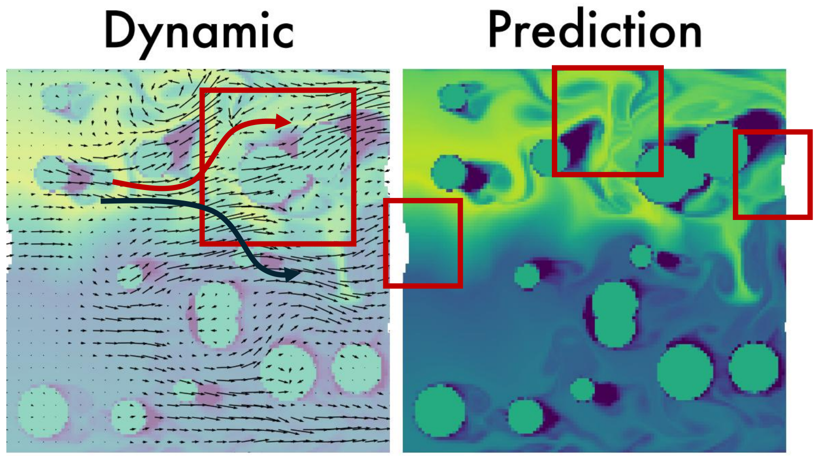
Highly rely on exact physics equations

Wande, et al. Learning Incompressible Fluid Dynamics from Scratch--Towards Fast, Differentiable Fluid Models that Generalize. ICLR 2021

Fluid Dynamics Modeling

Deep Optical flow:
Estimate fluid dynamics and predict future fluid field

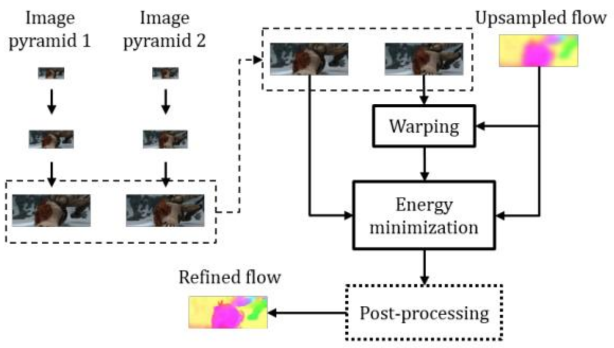


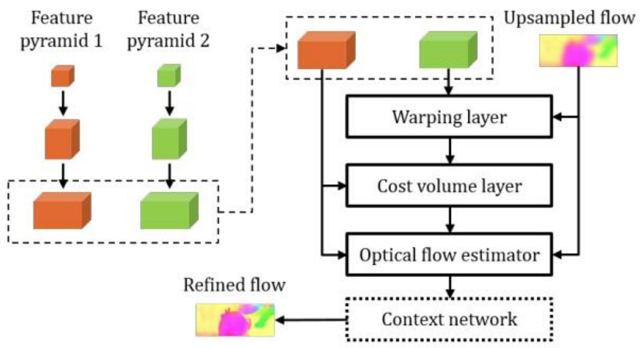


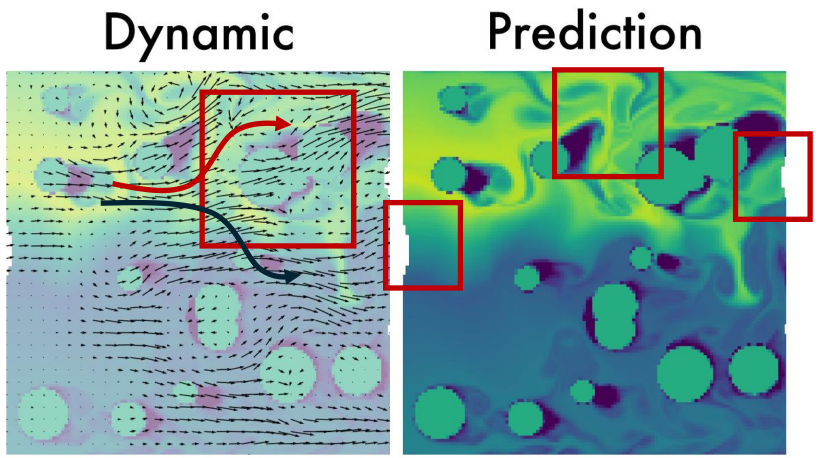
Sun, et al. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. CVPR 2018

Fluid Dynamics Modeling

Deep Optical flow:
Estimate fluid dynamics and predict future fluid field







Hard to capture complex dynamics

Sun, et al. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. CVPR 2018

Helmholtz Dynamics

From Helmholtz decomposition to *Helmholtz Dynamics*:
A 3D dynamic field can be decomposed into a curl-free component and a divergence-free component.

$$\mathbf{F}(\mathbf{r}) = \nabla\Phi(\mathbf{r}) + \nabla \times \mathbf{A}(\mathbf{r}),\ \mathbf{r} \in \mathbb{V}.$$

Helmholtz Dynamics

From Helmholtz decomposition to *Helmholtz Dynamics*:
A 3D dynamic field can be decomposed into a curl-free component and a divergence-free component.

$$\mathbf{F}(\mathbf{r}) = \nabla\Phi(\mathbf{r}) + \nabla \times \mathbf{A}(\mathbf{r}),\ \mathbf{r} \in \mathbb{V}.$$

↓

$$\mathbf{F}_{\text{Helm}}(\Phi, \mathbf{A}) = \nabla\Phi + \nabla \times \mathbf{A}$$
$$= \underbrace{\left(\frac{\partial\Phi}{\partial x}, \frac{\partial\Phi}{\partial y}\right)}_{\text{Curl-free Velocity}} + \underbrace{\left(\frac{\partial\mathbf{A}}{\partial y}, -\frac{\partial\mathbf{A}}{\partial x}\right)}_{\text{Divergence-free Velocity}}$$

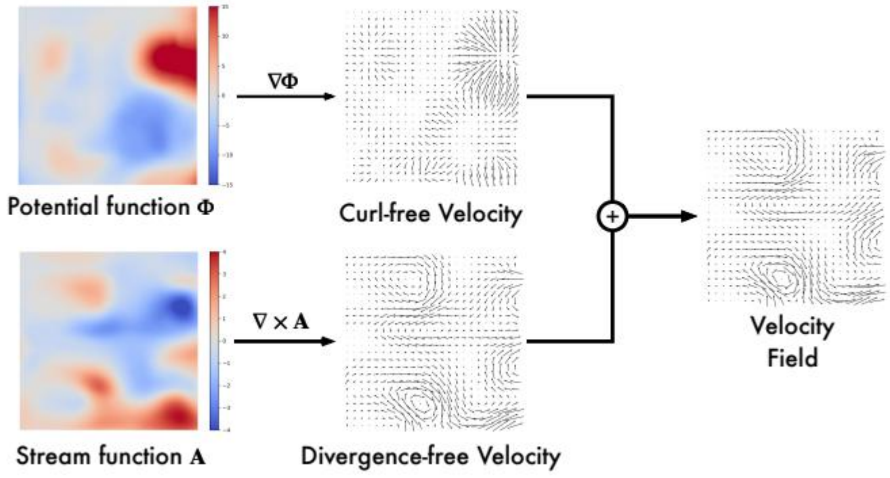
Helmholtz Dynamics

From Helmholtz decomposition to *Helmholtz Dynamics*:
A 3D dynamic field can be decomposed into a curl-free component and a divergence-free component.

$$\mathbf{F}(\mathbf{r}) = \nabla\Phi(\mathbf{r}) + \nabla \times \mathbf{A}(\mathbf{r}),\ \mathbf{r} \in \mathbb{V}.$$

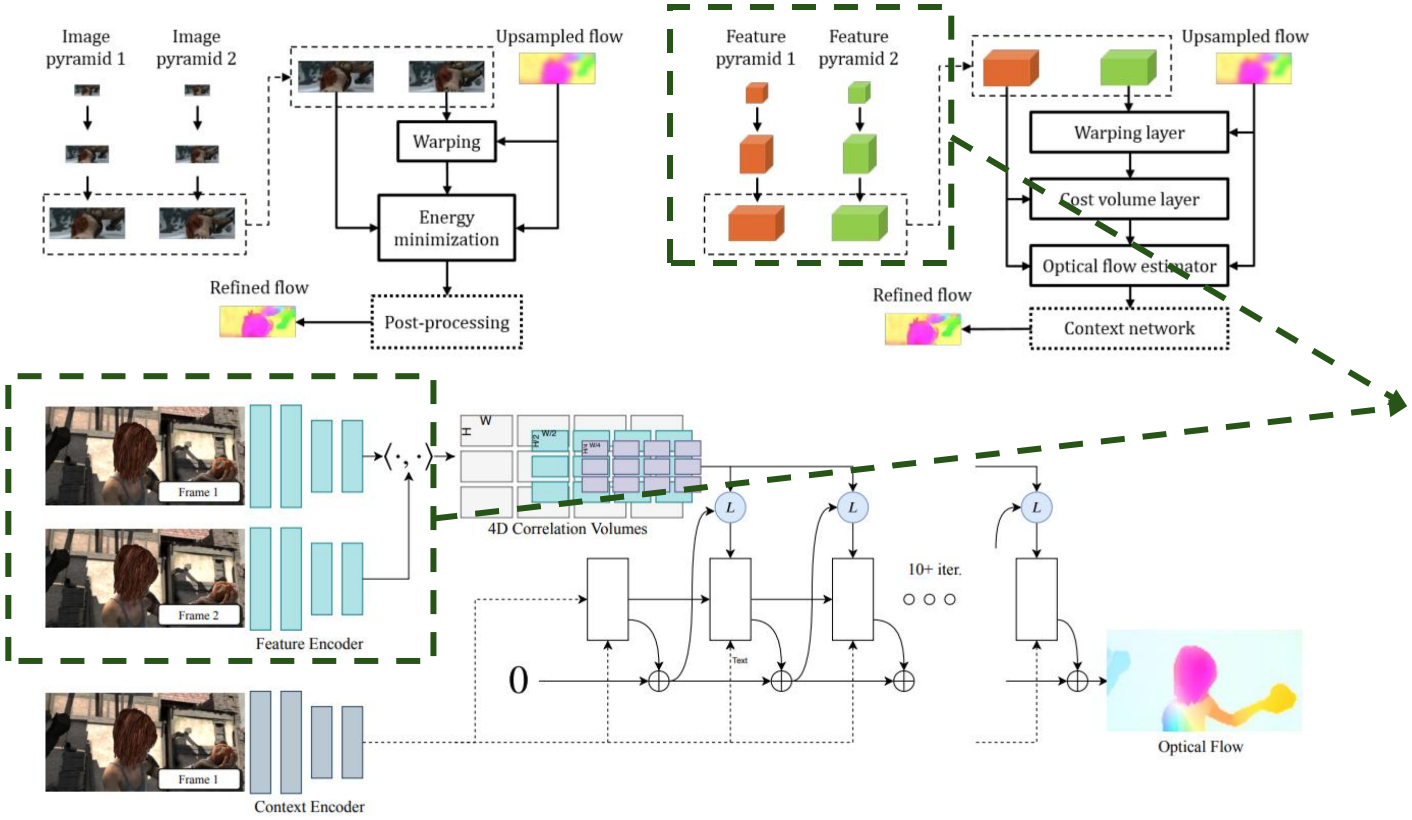
$$\mathbf{F}_{\text{Helm}}(\Phi, \mathbf{A}) = \nabla\Phi + \nabla \times \mathbf{A}$$

$$= \underbrace{\left(\frac{\partial\Phi}{\partial x}, \frac{\partial\Phi}{\partial y}\right)}_{\text{Curl-free Velocity}} + \underbrace{\left(\frac{\partial\mathbf{A}}{\partial y}, -\frac{\partial\mathbf{A}}{\partial x}\right)}_{\text{Divergence-free Velocity}}$$



HelmDynamics Block

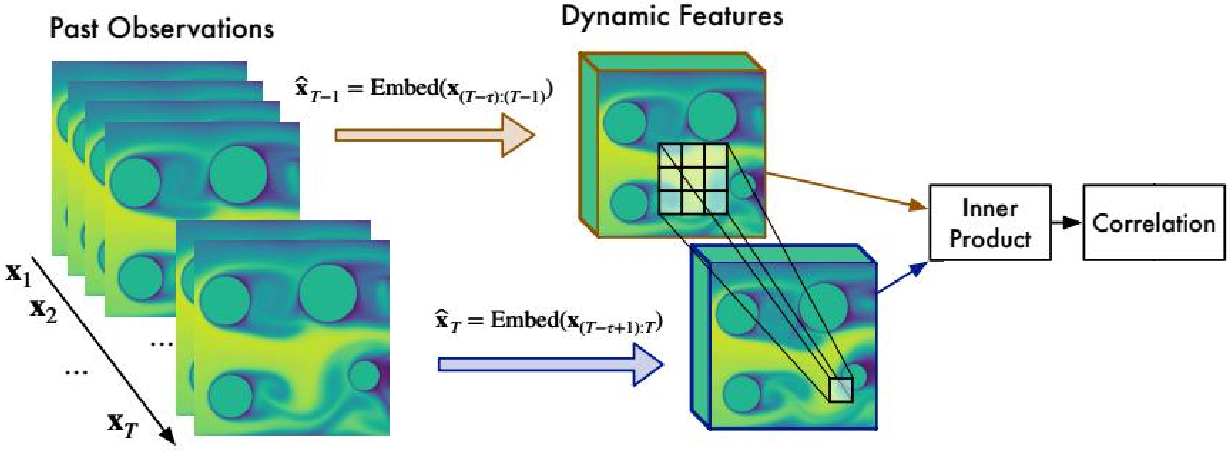
Recap: dynamics modeling



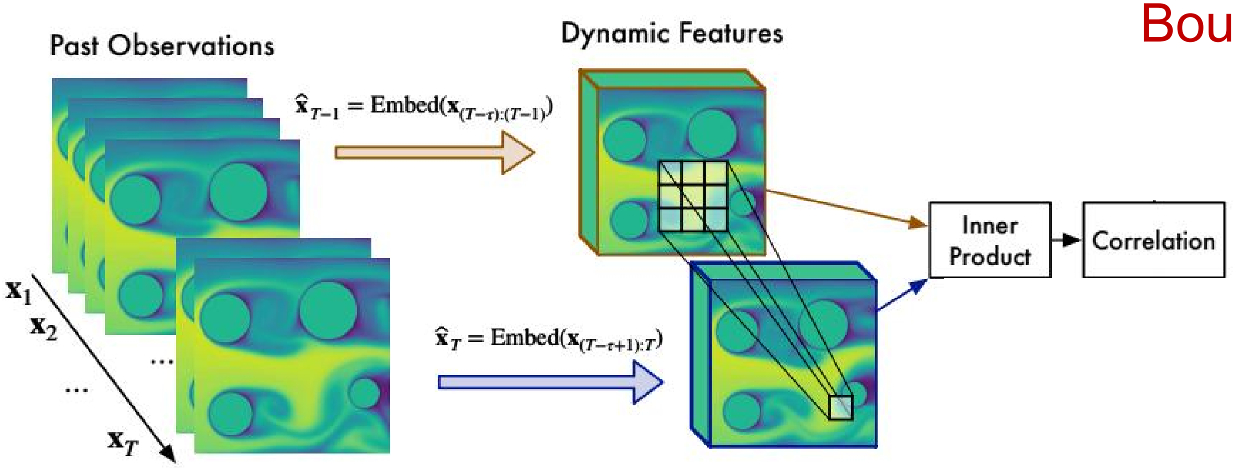
From correlation between adjacent frames to calculate velocity fields

Teed, et al. Raft: Recurrent all-pairs field transforms for optical flow. ECCV 2020

HelmDynamics Block



HelmDynamics Block

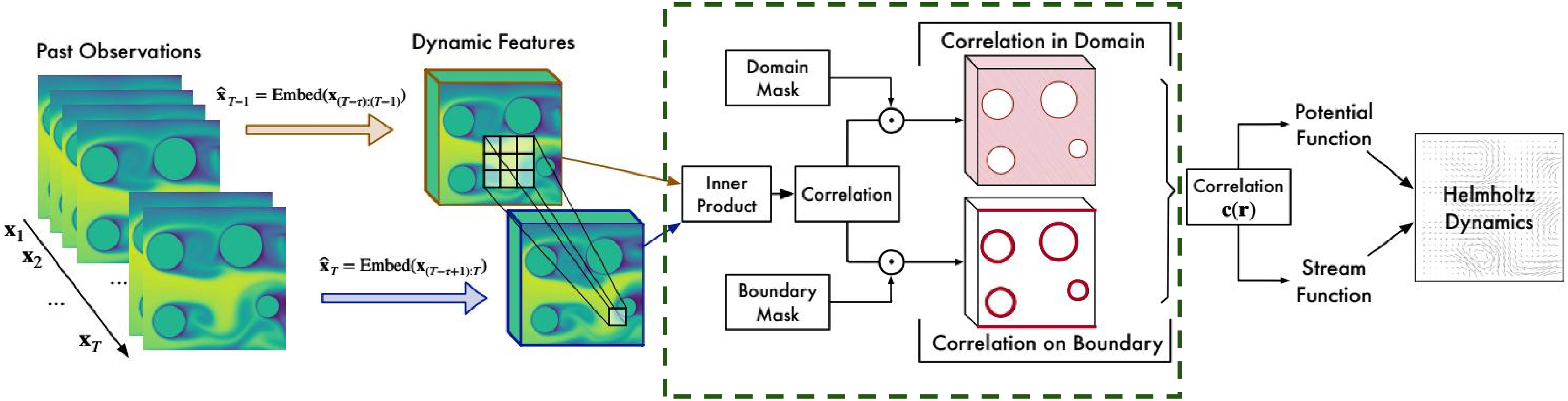


Boundary conditions in Helmholtz decomposition

$$\Phi(\mathbf{r}) \equiv \frac{1}{4\pi} \int_V \frac{\nabla' \cdot \mathbf{F}(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \mathrm{d}V' - \frac{1}{4\pi} \oint_S \hat{\mathbf{n}}' \cdot \frac{\mathbf{F}(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \mathrm{d}S'$$

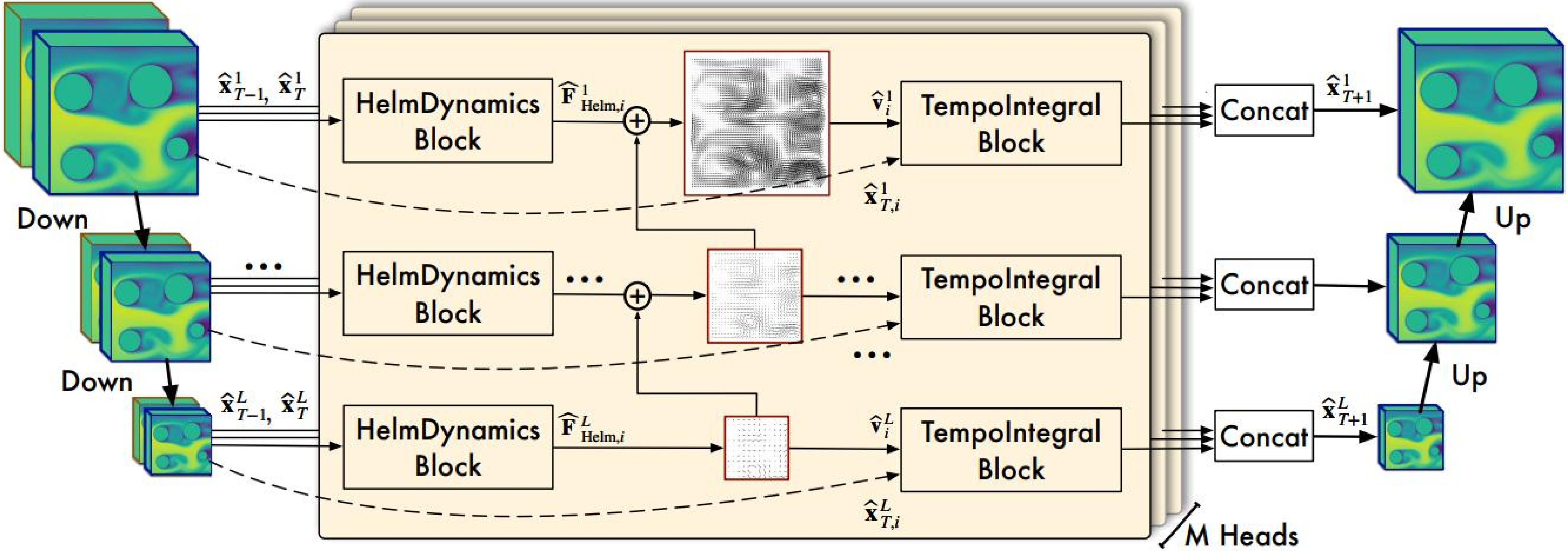
$$\mathbf{A}(\mathbf{r}) \equiv \frac{1}{4\pi} \int_V \frac{\nabla' \times \mathbf{F}(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \mathrm{d}V' - \frac{1}{4\pi} \oint_S \hat{\mathbf{n}}' \times \frac{\mathbf{F}(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \mathrm{d}S'$$

HelmDynamics Block

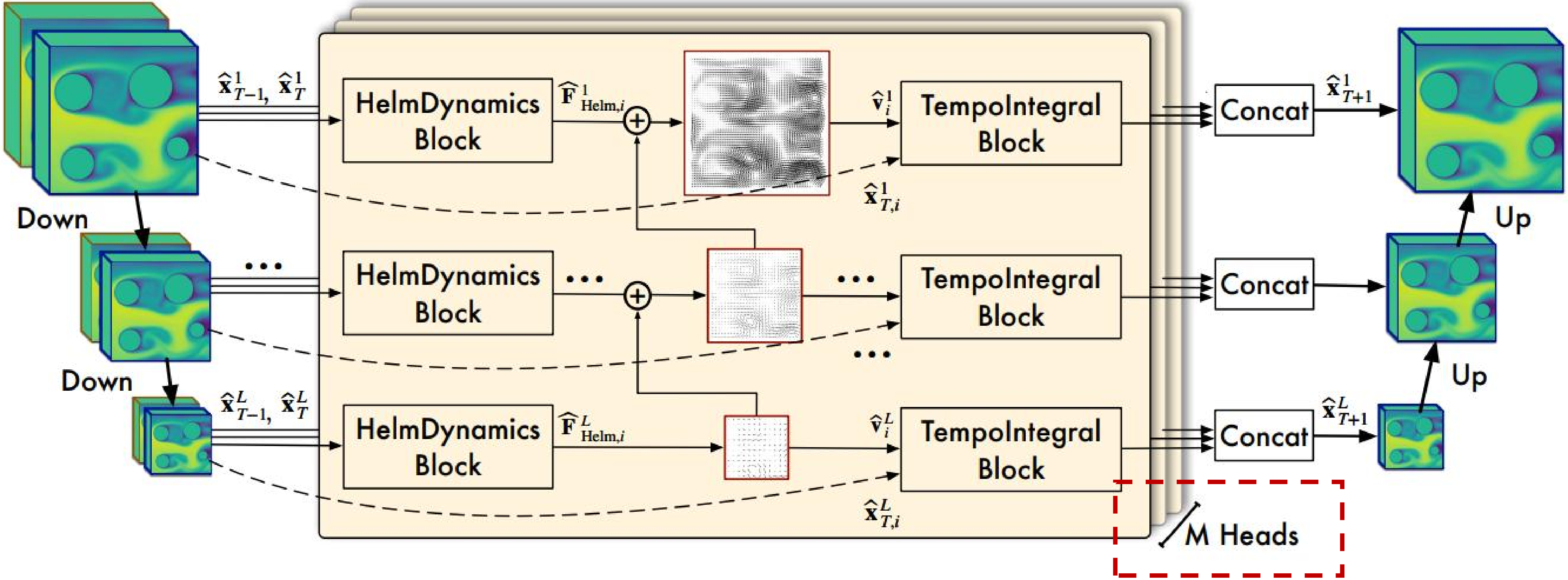


HelmDynamics block, which learns spatiotemporal correlations **c**(**r**) both **in the domain** and **on the boundary** to estimate potential and stream functions of fluid from past observations for composing the Helmholtz dynamics.

Multiscale Multihead Integral Architecture

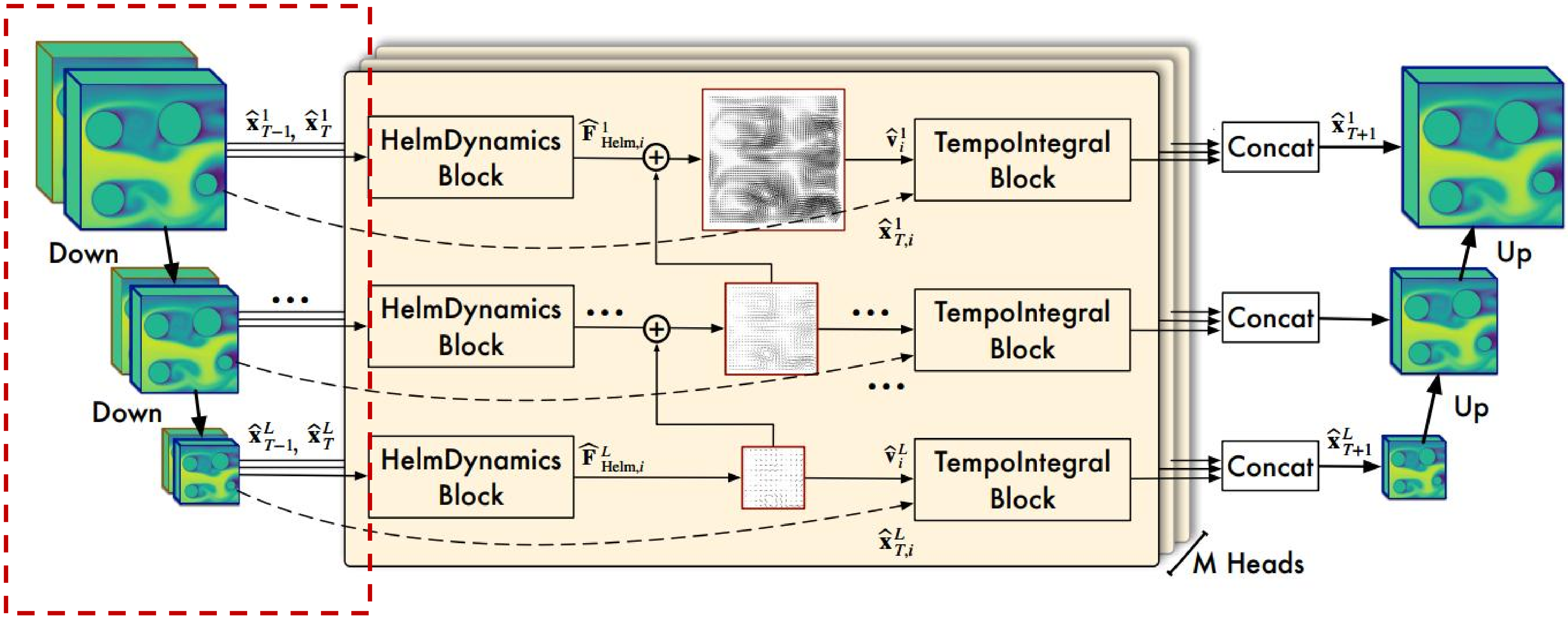


Multiscale Multihead Integral Architecture



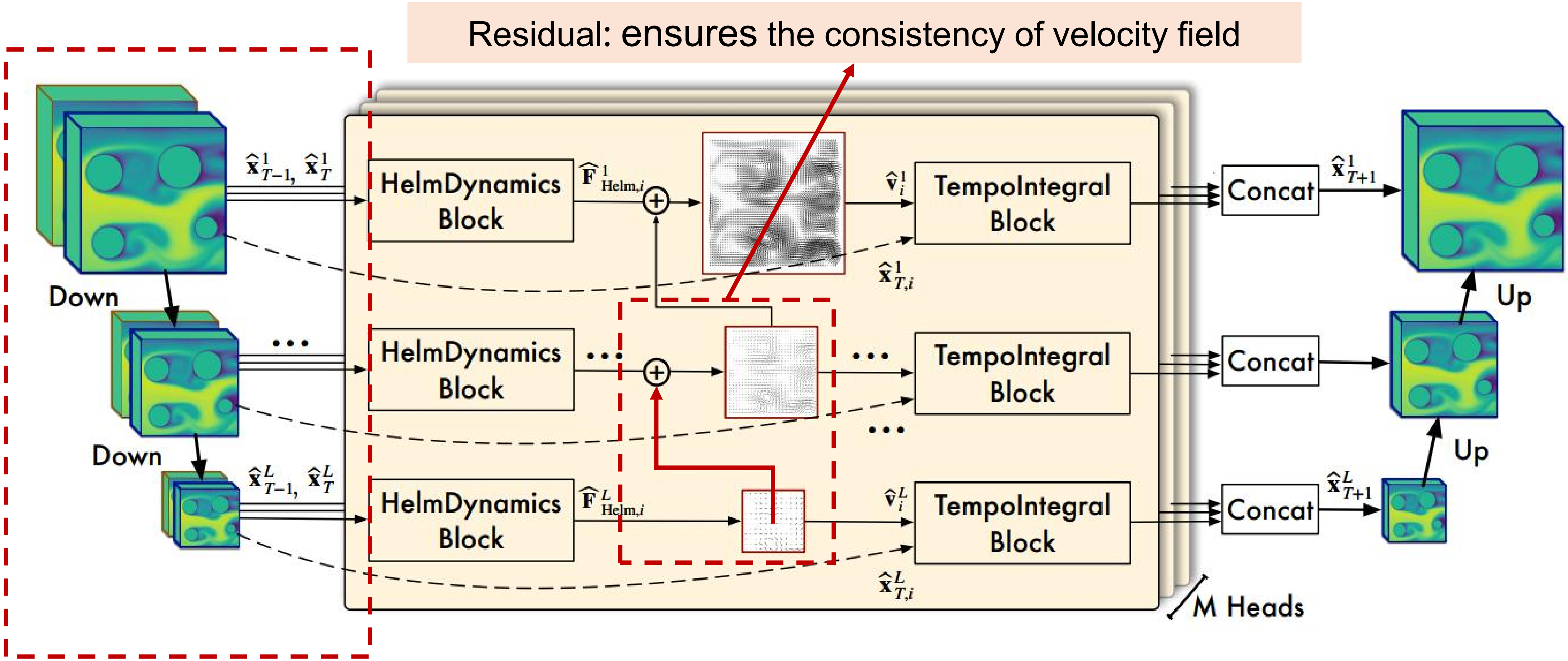
Multihead: capture different dynamic patterns

Multiscale Multihead Integral Architecture



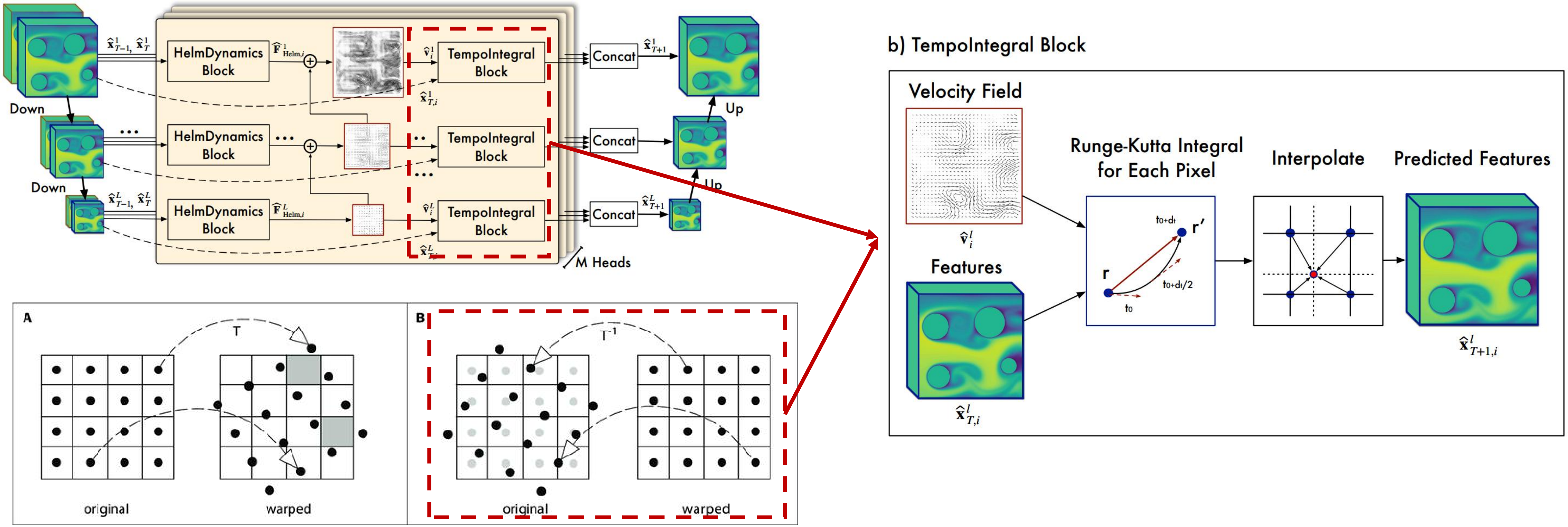
Multiscale: capture different properties at different scales

Multiscale Multihead Integral Architecture



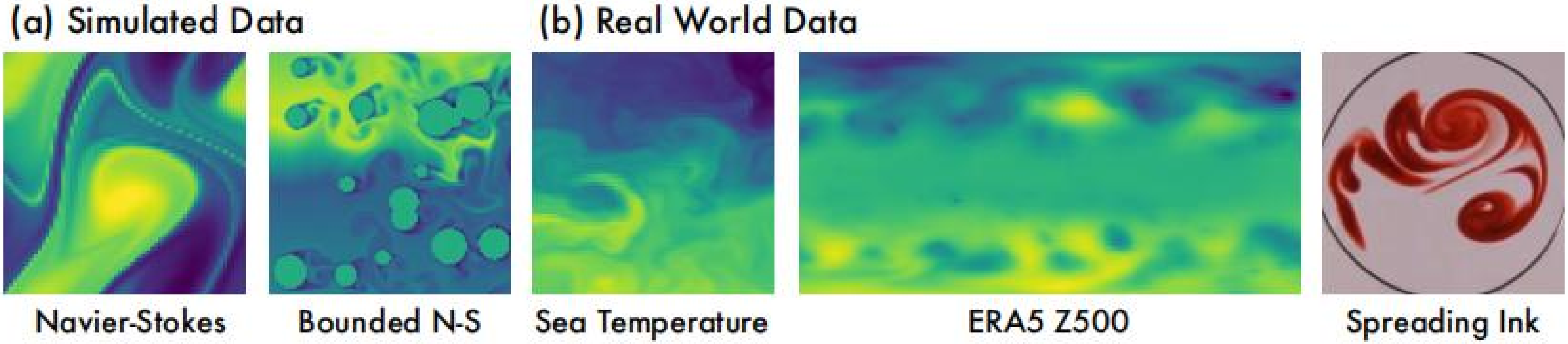
Multiscale: capture different properties at different scales

Multiscale Multihead Integral Architecture



We adopt the back-and-forth error compensation and correction (BFECC, (Kim et al., 2005)) for better position mapping

HelmFluid Experiments

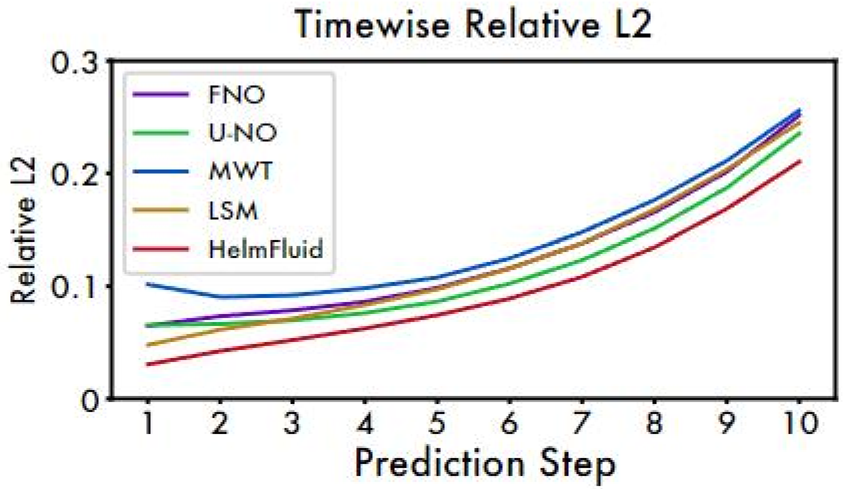


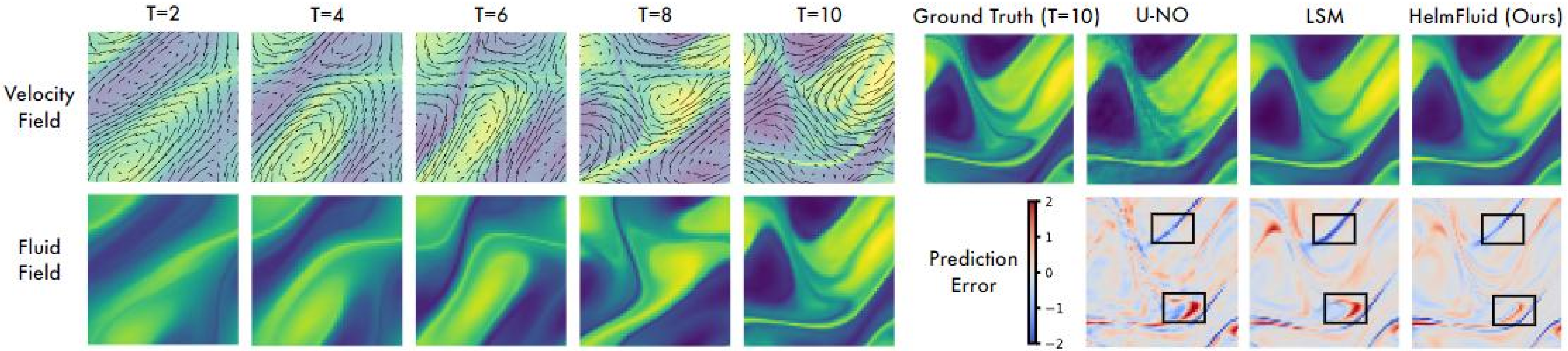
Dataset	(Input, predict length)	(Training, validation, test)	Observed state	Reynold numbers
Navier-Stokes	(10,10)	(1000,200,200)	Vorticity	$\sim 10^4$
Bounded N-S	(10,10)	(1000,200,200)	Grayscale	~ 300
ERA5 Z500	(2,10)	(20425, 2087, 4174)	Geopotential	Unknown
Sea Temperature	(10,10)	(170249, 17758, 65286)	Temperature	Unknown
Spreading Ink video 1	(100, 50)	One video sequence	RGB Image	Unknown
Spreading Ink video 2	(126, 63)	One video sequence	RGB Image	Unknown
Spreading Ink video 3	(93, 46)	One video sequence	RGB Image	Unknown

Simulated Data

Navier-Stokes dataset

Model	64 × 64	128 × 128	256 × 256
DARTS (Ruzanski et al., 2011)	0.8046	0.7002	0.7904
U-Net (Ronneberger et al., 2015)	0.1982	0.1589	0.2953
FNO (Li et al., 2021)	0.1556	0.1028	0.1645
MWT (Gupta et al., 2021)	0.1586	0.0841	0.1390
U-NO (Rahman et al., 2023)	0.1435	0.0913	0.1392
LSM (Wu et al., 2023)	0.1535	0.0961	0.1973
HelmFluid (Ours)	**0.1261**	**0.0807**	**0.1310**
Promotion	12.1%	4.0%	5.8%

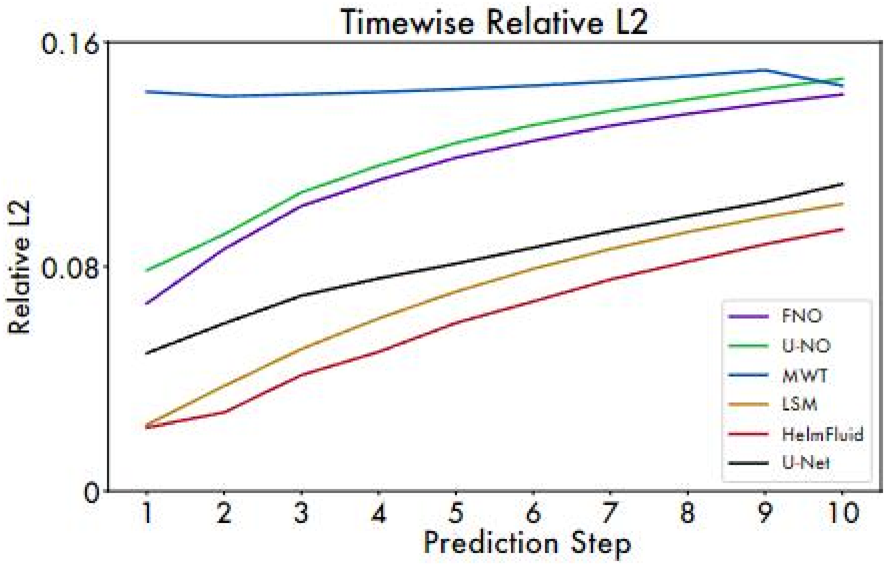


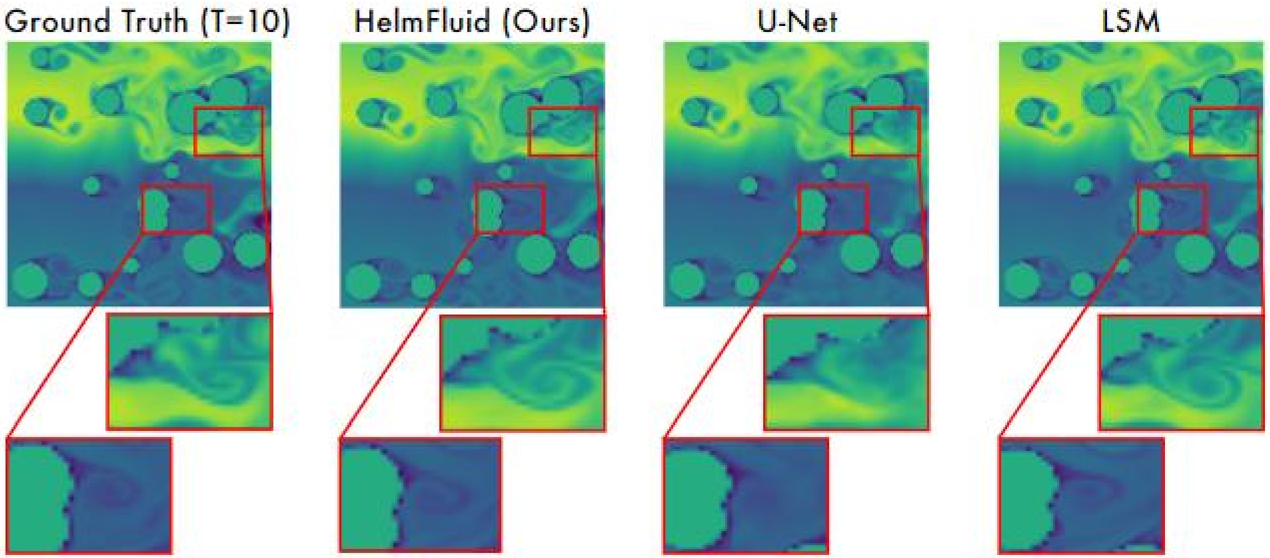


Simulated Data

Bounded N-S dataset

Model	Relative L2
DARTS (Ruzanski et al., 2011)	0.1820
U-Net (Ronneberger et al., 2015)	0.0846
FNO (Li et al., 2021)	0.1176
MWT (Gupta et al., 2021)	0.1407
U-NO (Rahman et al., 2023)	0.1200
LSM (Wu et al., 2023)	<u>0.0737</u>
HelmFluid (Ours)	**0.0652**
Promotion	11.5%

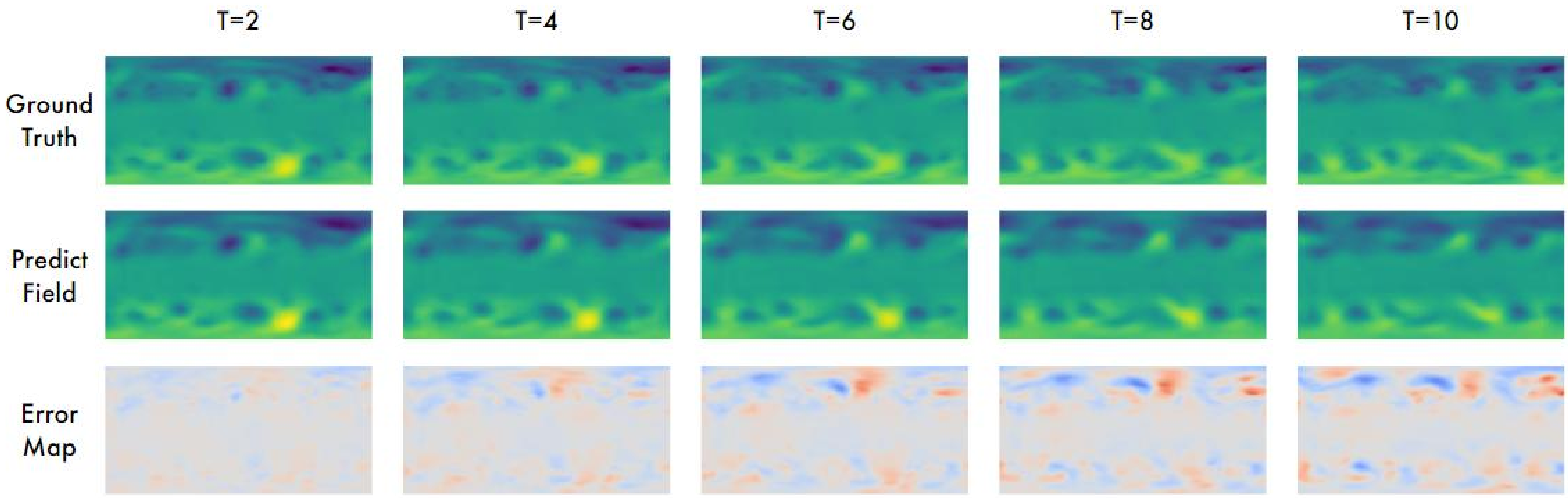




Real-World Data

ERA5 Z500 dataset

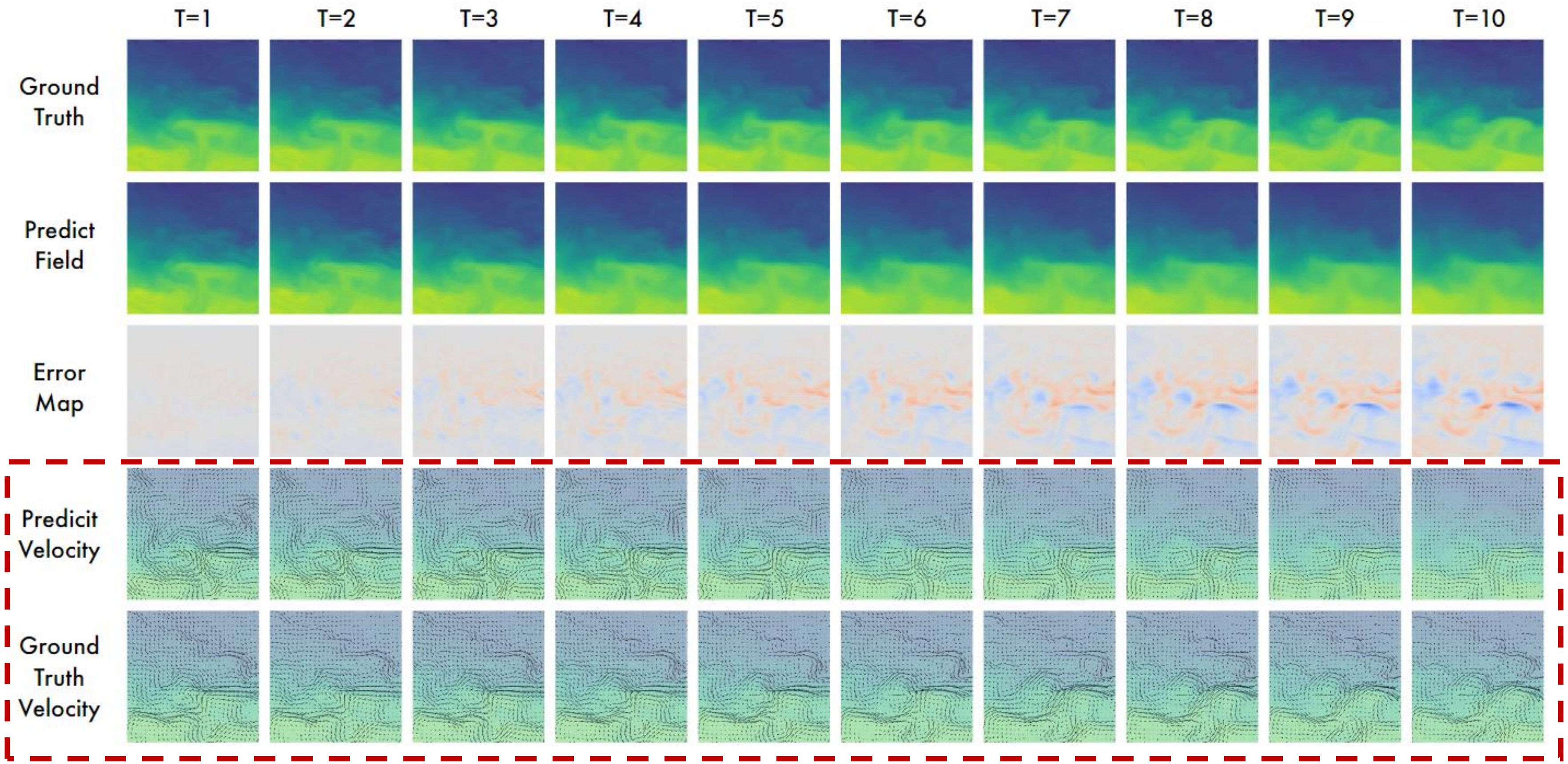
Models	RMSE
U-Net (Ronneberger et al., 2015)	632.94
FNO (Li et al., 2021)	596.80
MWT (Gupta et al., 2021)	596.45
U-NO (Rahman et al., 2023)	596.84
LSM (Wu et al., 2023)	561.27
FourcastNet (Pathak et al., 2022)	594.49
HelmFluid (ours)	**521.44**
Promotion	7.1%



Real-World Data

Sea Temperature dataset

Models	Relative L2	MSE
DARTS (Ruzanski et al., 2011)	0.3308	0.1094
U-Net (Ronneberger et al., 2015)	0.1735	0.0379
FNO (Li et al., 2021)	0.1935	0.0456
MWT (Gupta et al., 2021)	0.2075	0.0506
U-NO (Rahman et al., 2023)	0.1969	0.0472
LSM (Wu et al., 2023)	0.1759	0.0389
HelmFluid (Ours)	**0.1704**	**0.0368**
Promotion	1.8%	2.9%



Real-World Data

Spreading Ink dataset

MODELS	METRICS
U-NET (RONNEBERGER ET AL., 2015)	3.596 / 0.2620 / 0.0176
FNO (LI ET AL., 2021)	4.095 / 0.2776 / 0.0198
U-NO (RAHMAN ET AL., 2023)	5.604 / 0.2971 / 0.0227
VORTEX (DENG ET AL., 2023)	3.949 / 0.2483 / 0.0161
LSM (WU ET AL., 2023)	3.760 / 0.2698 / 0.0187
HELMFLUID (OURS)	**3.323 / 0.2183 / 0.0125**
PROMOTION	7.6% / 12.1% / 22.3%

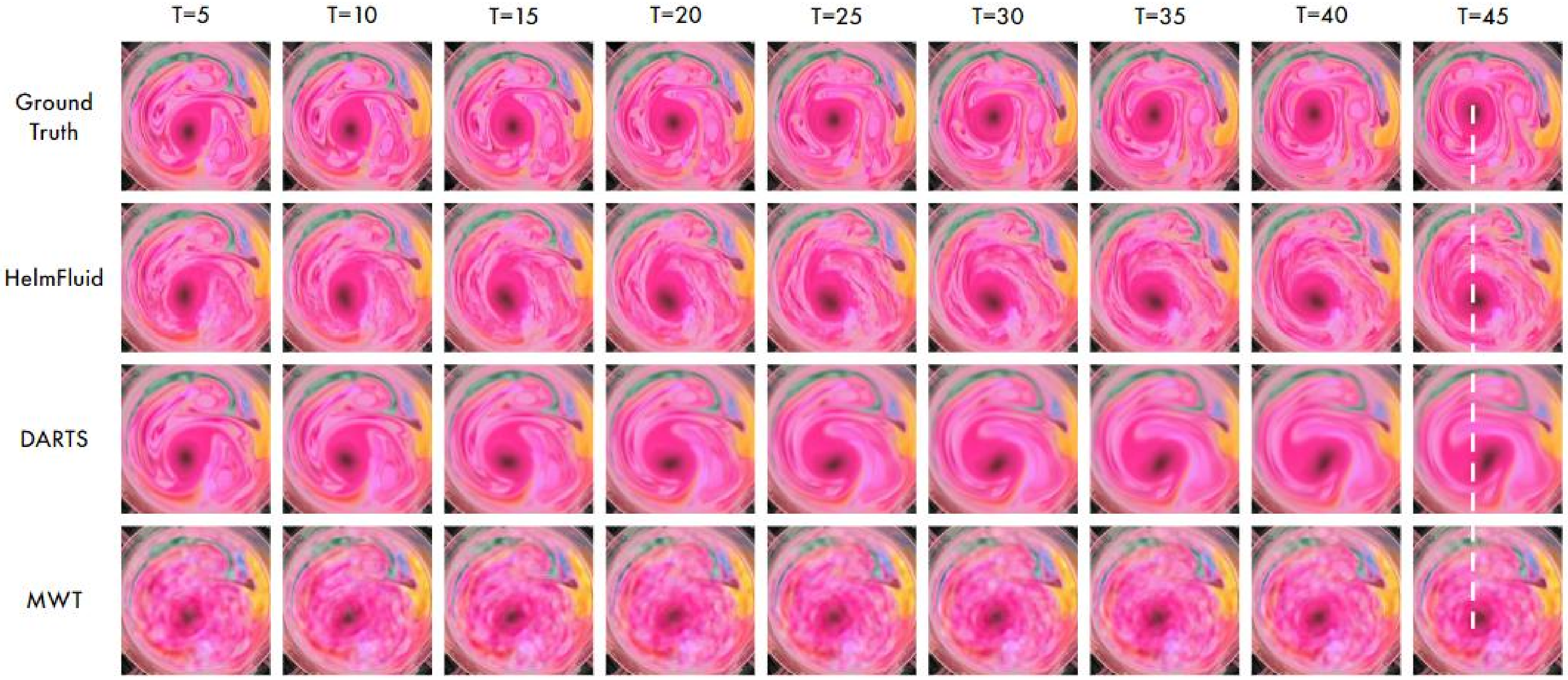
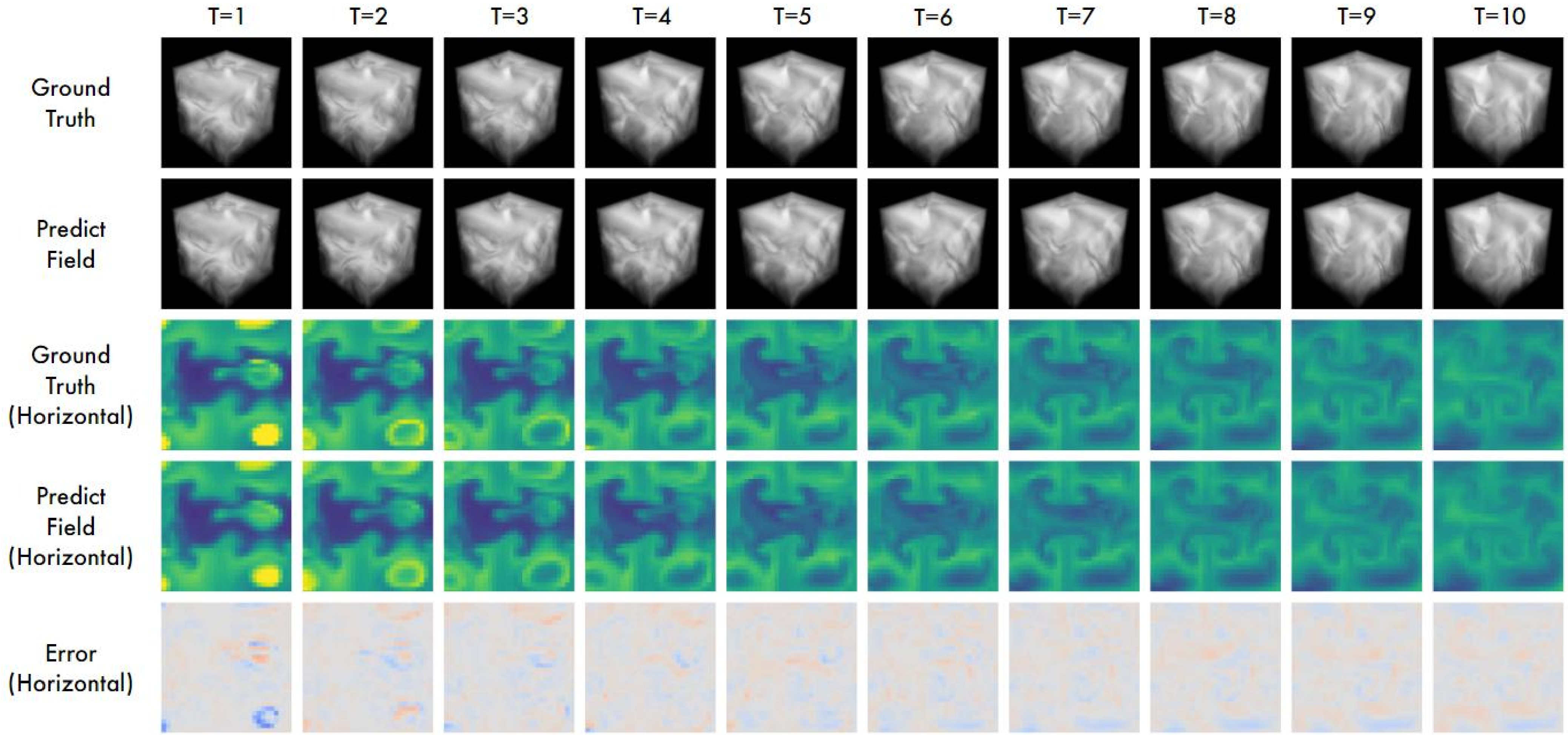


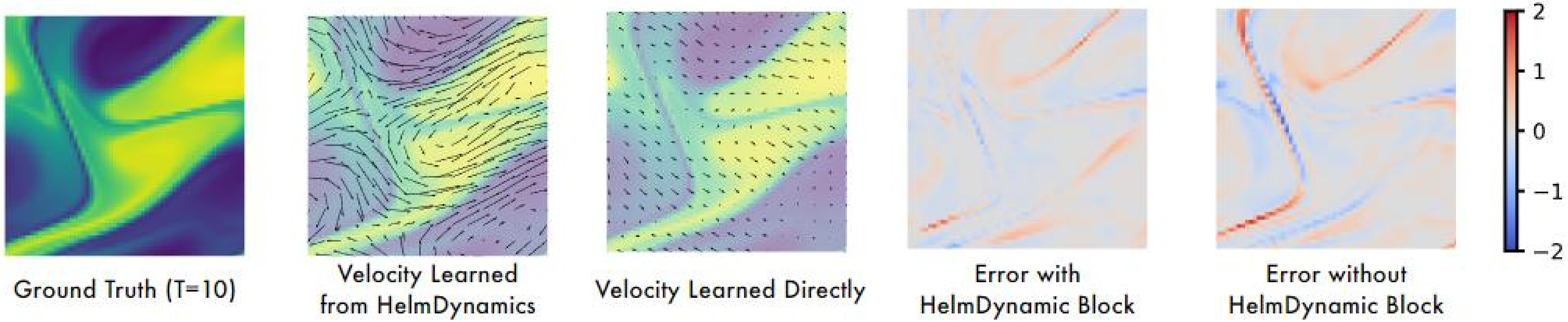
Figure 15. Showcases of HelmFluid, DARTS, and MWT on the Spread Ink dataset .

Extend to 3D


T=1
T=2
T=3
T=4
T=5
T=6
T=7
T=8
T=9
T=10
Ground Truth
Predict Field
Ground Truth (Horizontal)
Predict Field (Horizontal)
Error (Horizontal)

Ablations

With / without HelmDynamics



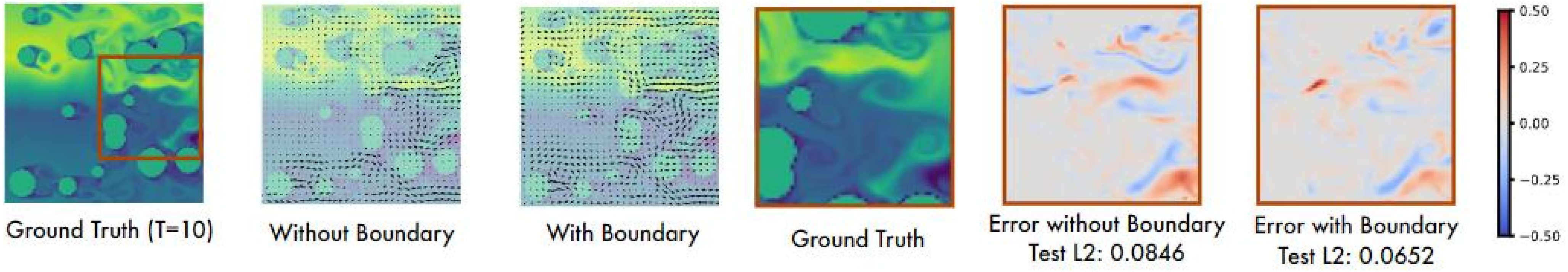
Relative L2: 0.1261 0.1412

With / without potential / stream function

Metrics	HelmDynamics	Only potential function	Only stream function
Relative L2	**0.1261**	0.1460	0.1305
GPU memory (GB)	16.30	16.29	16.30
Training Time (s / epoch)	80.20	79.57	79.60

Ablations

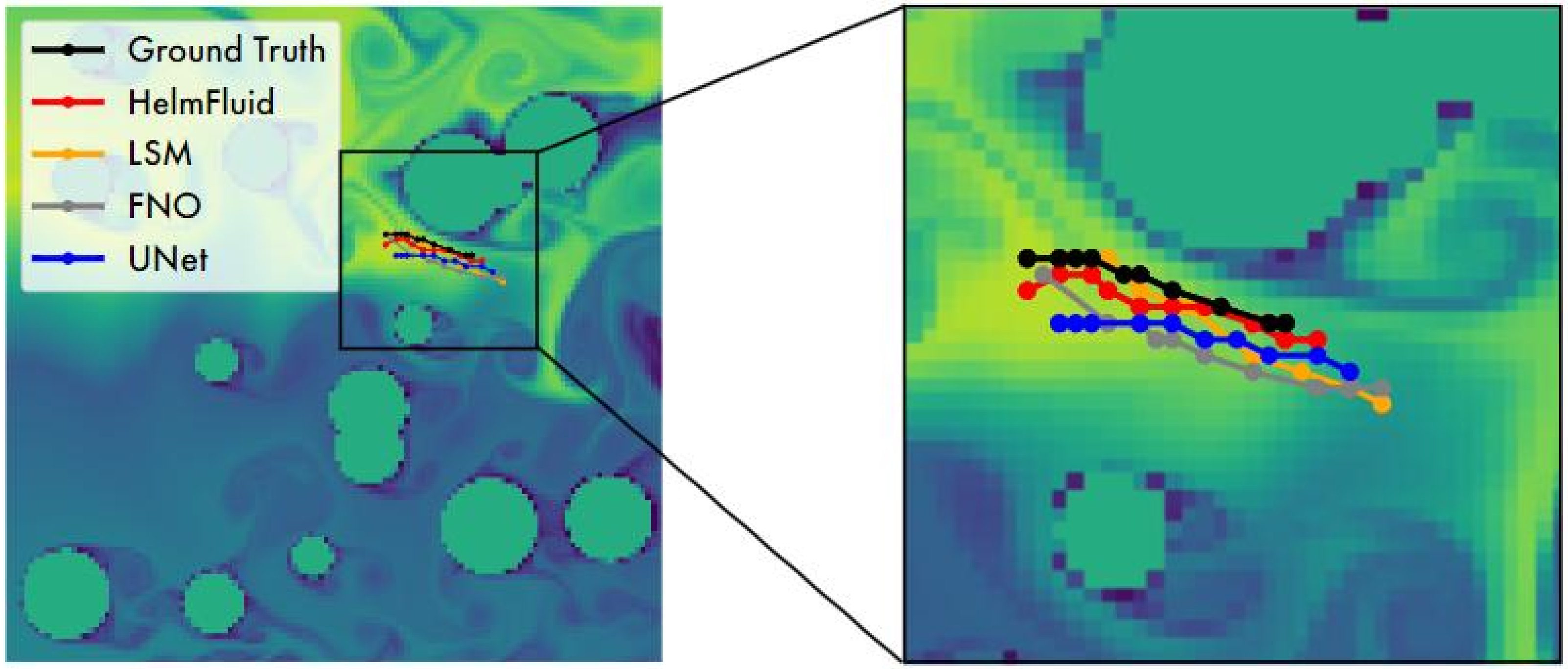
With / without boundary conditions



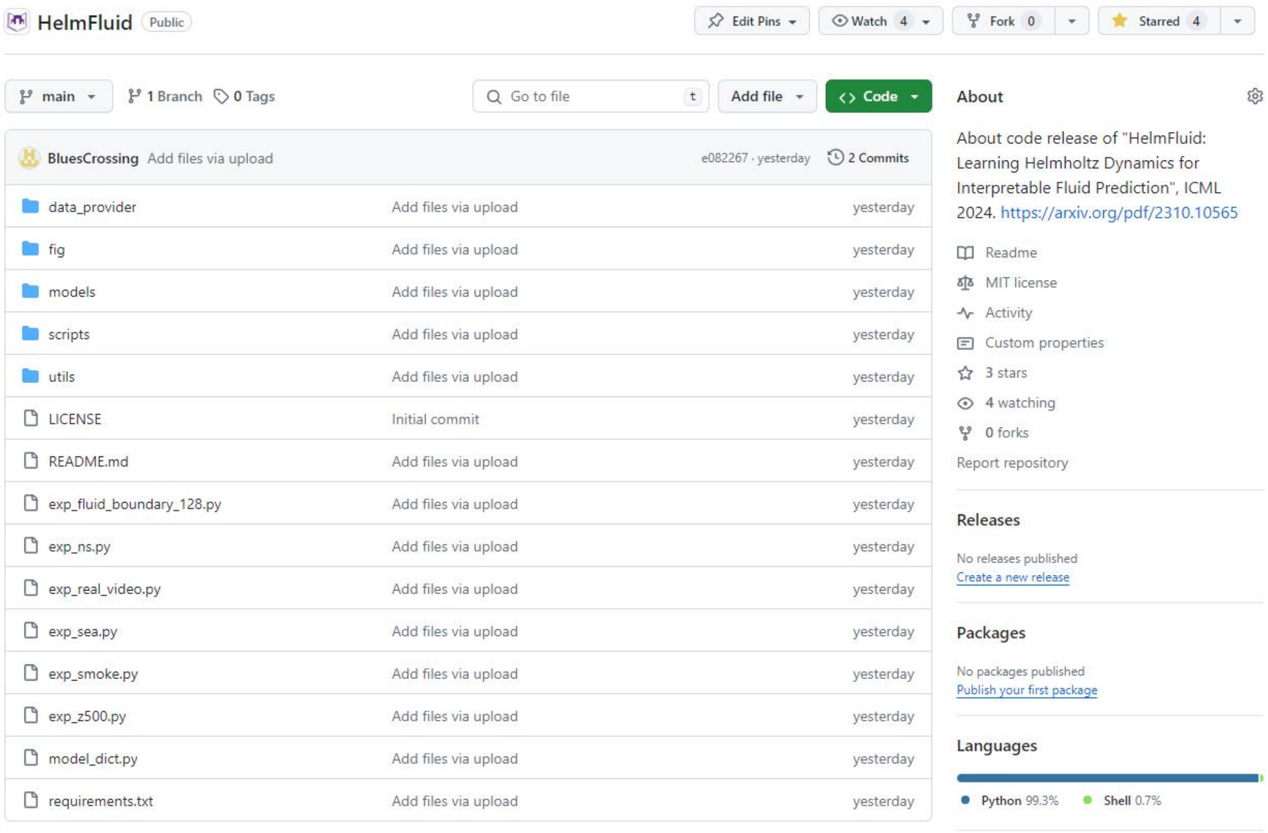
Different hyperparameter

Order of Runge-Kutta	1	2	3	4
Relative L2	0.1298	**0.1261**	0.1268	0.1278
Training time (s / epoch)	80.04	81.20	88.30	90.49
Number of heads M	1	4	8	16
Parameter number	11,063,245	9,929,101	9,812,653	9,762,205
Relative L2	0.1344	<u>0.1261</u>	0.1279	**0.1249**
Training time (s / epoch)	59.69	81.20	120.86	171.97
Number of scales L	2	3	4	5
Parameter number	9,283,977	9,929,101	15,906,193	29,820,309
Relative L2	0.1514	**0.1261**	0.1361	<u>0.1330</u>
Training time (s / epoch)	64.43	81.20	99.83	120.06

Dynamics Tracking



Open Source



https://github.com/thuml/HelmFluid

Thank You!

xlx22@mails.tsinghua.edu.cn

长按关注，获取最新资讯