Physics-informed Neural Networks for Super-Resolution of Turbulent Flows

Super Resolution with Data and Physics

Diego Di Carlo

July 5, 2022

Supervisor Thomas Corpetti (Univ. Rennes 2)

Advisors: Nicolas Courty (Univ. Bretagne Sud), Dominique Heitz (INRAE Rennes)

DynaLearn Project, Université de Rennes 2, LETG, CNRS

MACLEAN Workshop
CAP&RFIAP 2022

Super Resolution = increase resolution

- Images → increase the *spatial* resolution (∼ upsampling, enhancement)
- Audio → increase *temporal* resolution (∼ bandwidth extension)
- Remote sensing data → increase *spatial and temporal* resolution of such data

Super Resolution = increase resolution

- Images → increase the *spatial* resolution (∼ upsampling, enhancement)
- Audio → increase *temporal* resolution (∼ bandwidth extension)
- Remote sensing data → increase *spatial and temporal* resolution of such data

Resolution **loss** possibly due to ...

- **measurement** process (e.g. sensors bandwidth, small number of sensors, noise)
- lossy **compression** (e.g. MPEG coding)
- computational **complexity** (e.g. fast simulation)

Super Resolution = increase resolution

- Images → increase the *spatial* resolution (∼ upsampling, enhancement)
- Audio → increase *temporal* resolution (∼ bandwidth extension)
- Remote sensing data → increase *spatial and temporal* resolution of such data

Resolution **loss** possibly due to ...

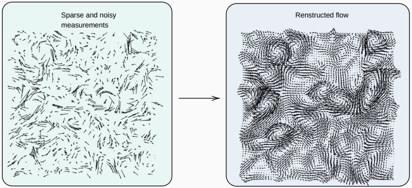
- **measurement** process (e.g. sensors bandwidth, small number of sensors, noise)
- lossy **compression** (e.g. MPEG coding)
- computational **complexity** (e.g. fast simulation)

It's **inverse ill-posed problem** → How to recover the missing information? And which?

Super Resolution of Turbulent flows

In particular, I work on

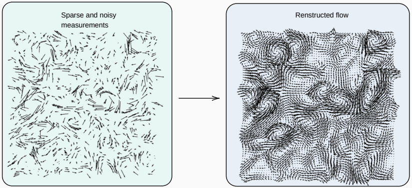
- Super-resolution ...
- ... of velocity fields ... (sparse, noisy)
- ... describing turbulences



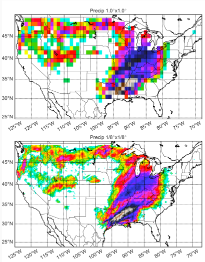
Super Resolution of Turbulent flows

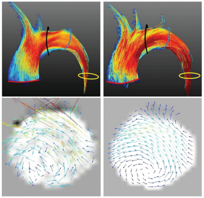
In particular, I work on

- Super-resolution ...
- ... of velocity fields ... (sparse, noisy)
- ... describing turbulences



Applications





Geo-science

Wind, Temperature, Pollution, *etc.*

Fluid Dynamics

Fluids simulation and identification

Biomedics

Blood flows

Model: Physics-informed Neural Networks (PINNs)

PINNs are DNNs that learn the solution of a PDE [Raissi et al., 2019]:

$$f(\mathbf{x}, t, \Phi, \nabla_{\mathbf{x}}\Phi, \nabla^2_{\mathbf{x}}\Phi, \partial_t\Phi, ...) = 0, \quad \Phi : \mathbf{x}, t \to \Phi(\mathbf{x}, t) \tag{1}$$

with $\mathbf{x} \in \Omega$, $t \in [0, T]$, and Φ a non-linear diff. operator (e.g. Navier-Stokes eq.)

Model: Physics-informed Neural Networks (PINNs)

PINNs are DNNs that learn the solution of a PDE [Raissi et al., 2019]:

$$f(\mathbf{x}, t, \Phi, \nabla_{\mathbf{x}}\Phi, \nabla^2_{\mathbf{x}}\Phi, \partial_t\Phi, ...) = 0, \quad \Phi : \mathbf{x}, t \rightarrow \Phi(\mathbf{x}, t) \tag{1}$$

with $\mathbf{x} \in \Omega$, $t \in [0, T]$, and Φ a non-linear diff. operator (e.g. Navier-Stokes eq.)

Core idea:

$$\begin{cases} \Phi(\mathbf{x}, t) & \approx \mathtt{DNN}(\mathbf{x}, t) \\ \partial_t\Phi(\mathbf{x}, t) & \approx \mathtt{autograd}_t\ \mathtt{DNN}(\mathbf{x}, t) \\ \nabla_{\mathbf{x}}, \nabla^2_{\mathbf{x}}, ... & \approx \mathtt{autograd}_{\mathbf{x}}\ \mathtt{DNN}(\mathbf{x}, t) \end{cases}$$

$$\mathcal{L} = \mathcal{L}_{\text{rec.}} + \mathcal{L}_{\text{PDE}}$$

$\mathcal{L}_{\text{rec.}}$ comprises MSE on observations, initial and boundary conditions

Model: Physics-informed Neural Networks (PINNs)

PINNs are DNNs that learn the solution of a PDE [Raissi et al., 2019]:

$$f(\mathbf{x}, t, \Phi, \nabla_{\mathbf{x}}\Phi, \nabla^2_{\mathbf{x}}\Phi, \partial_t\Phi, ...) = 0, \quad \Phi : \mathbf{x}, t \rightarrow \Phi(\mathbf{x}, t) \tag{1}$$

with $\mathbf{x} \in \Omega$, $t \in [0, T]$, and Φ a non-linear diff. operator (e.g. Navier-Stokes eq.)

Core idea:

$$\begin{cases} \Phi(\mathbf{x}, t) & \approx \texttt{DNN}(\mathbf{x}, t) \\ \partial_t\Phi(\mathbf{x}, t) & \approx \texttt{autograd}_t\ \texttt{DNN}(\mathbf{x}, t) \\ \nabla_{\mathbf{x}}, \nabla^2_{\mathbf{x}}, ... & \approx \texttt{autograd}_{\mathbf{x}}\ \texttt{DNN}(\mathbf{x}, t) \end{cases}$$

$$\mathcal{L} = \mathcal{L}_{\text{rec.}} + \mathcal{L}_{\text{PDE}}$$

$\mathcal{L}_{\text{rec.}}$ comprises MSE on observations, initial and boundary conditions

- Once trained, the DNN is able to **evaluate any new point** $\mathbf{x}^* \in \Omega$ and $t^* \in [0, T]$.
- It acts like **Kernel interpolation** (Neural Tangent Kernel [Tancik et al., 2020])

Model: Physics-informed Neural Networks (PINNs)

PINNs are DNNs that learn the solution of a PDE [Raissi et al., 2019]:

$$f(\mathbf{x}, t, \Phi, \nabla_{\mathbf{x}}\Phi, \nabla^2_{\mathbf{x}}\Phi, \partial_t\Phi, ...) = 0, \quad \Phi : \mathbf{x}, t \to \Phi(\mathbf{x}, t) \tag{1}$$

with $\mathbf{x} \in \Omega$, $t \in [0, T]$, and Φ a non-linear diff. operator (e.g. Navier-Stokes eq.)

Core idea:

$$\begin{cases} \Phi(\mathbf{x}, t) & \approx \mathtt{DNN}(\mathbf{x}, t) \\ \partial_t\Phi(\mathbf{x}, t) & \approx \mathtt{autograd}_t\ \mathtt{DNN}(\mathbf{x}, t) \\ \nabla_{\mathbf{x}}, \nabla^2_{\mathbf{x}}, ... & \approx \mathtt{autograd}_{\mathbf{x}}\ \mathtt{DNN}(\mathbf{x}, t) \end{cases}$$

$$\mathcal{L} = \mathcal{L}_{\text{rec.}} + \mathcal{L}_{\text{PDE}}$$

$\mathcal{L}_{\text{rec.}}$ comprises MSE on observations, initial and boundary conditions

- Once trained, the DNN is able to **evaluate any new point** $\mathbf{x}^* \in \Omega$ and $t^* \in [0, T]$.
- It acts like **Kernel interpolation** (Neural Tangent Kernel [Tancik et al., 2020])

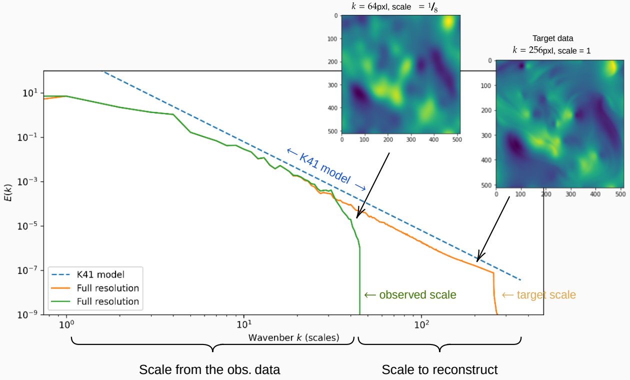
✓Unsupervised ✓Meshless ✓Data and Model driven ✗PDE terms only as regularizers

Proposed Approach: PINNs for Unsupervised Super-resolution

Low resolution data + **Physical models**

Low resolution data ↓ reconstruction loss / initial condition

Physical models ↓ Regularize higher unseen data

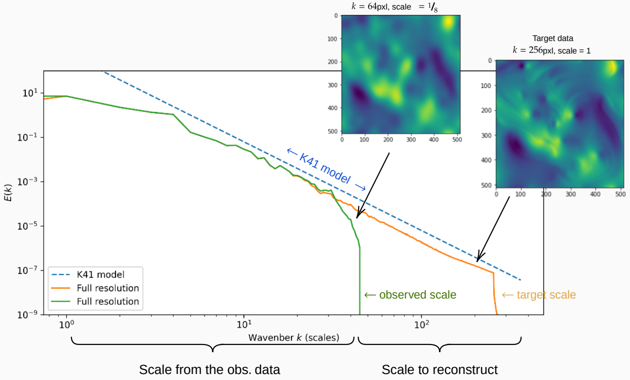


Proposed Approach: PINNs for Unsupervised Super-resolution

Low resolution data ↓ reconstruction loss / initial condition

+

Physical models ↓ Regularize higher unseen data



Physical models:

Divergence-free output

Sub-grid models (Structure fun.)

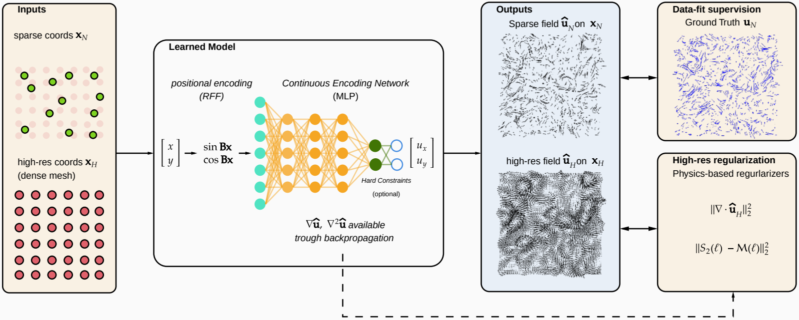
Smooth gradient regularization

Navier-Stokes PDE (if temporal data)

Proposed Approach: pipeline

Super-resolution of **Instantaneous** Velocity Field (= just a single snapshot)

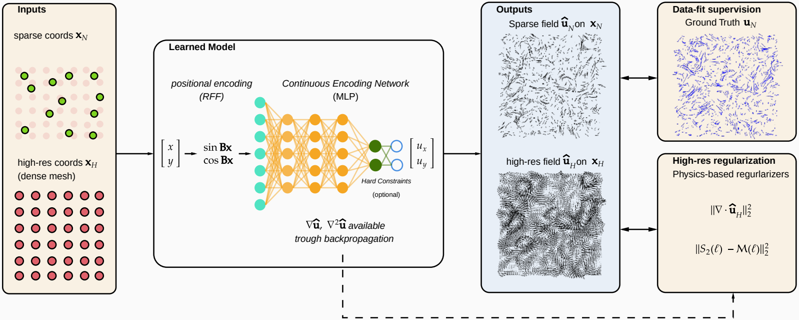
- Only **divergence-free** $\rightarrow$ soft or hard constraints
- Navier-Stokes need temporal evolution $\rightarrow$ **no PDE** available
- **Structure functions** as sub-grid model [Effinger and Grossmann, 1987]



Proposed Approach: pipeline

Super-resolution of **Instantaneous** Velocity Field (= just a single snapshot)

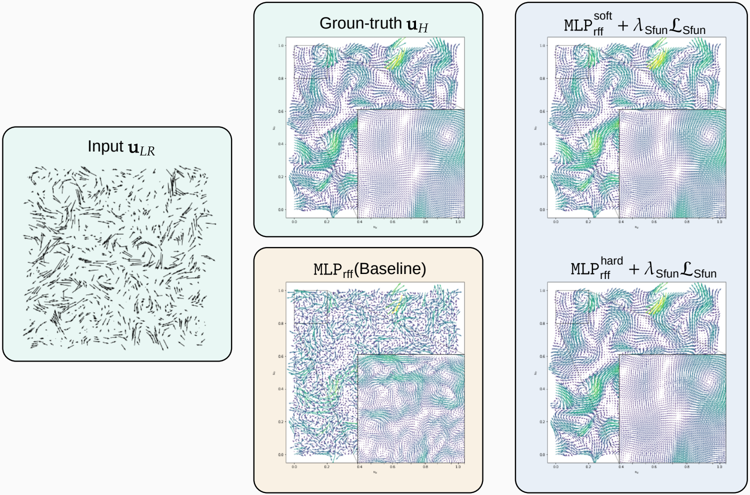
- Only **divergence-free** → soft or hard constraints
- Navier-Stokes need temporal evolution → **no PDE** available
- **Structure functions** as sub-grid model [Effinger and Grossmann, 1987]



⚠ Not "image to image", but **coordinate to "image"**

⚠ Not dataset of "images", but **dataset** of pixels of **one image**

Proposed Approach: Qualitative results



Quantitative results published in *20th International Symposium on Applications of Laser and Imaging Techniques to Fluid Mechanics, Lisbon, 2022*

Conclusion and Current Work

Take home messages for PINNs

- ✓ **Best of the data- and model-driven approach**
- ✓ Able to perform **unsupervised meshless evaluation** (vs. classical CNN)
- ✓ **Custom prior physical knowledge** (e.g., sub-grid models) as regularizers
- ✗ Some **hard constraint** can be implemented → ⚠ artifacts on HR and gradients
- ✗ Difficulty to minimize multi-task learning objectives ($*$)

Conclusion and Current Work

Take home messages for PINNs

- ✓ **Best of the data- and model-driven approach**
- ✓ Able to perform **unsupervised meshless evaluation** (vs. classical CNN)
- ✓ **Custom prior physical knowledge** (e.g., sub-grid models) as regularizers
- ✗ Some **hard constraint** can be implemented → ⚠ artifacts on HR and gradients
- ✗ Difficulty to minimize multi-task learning objectives (∗)

Links to other research fields

- to **Implicit Neural Representation** (**NeRF**, SIREN, BACON [Lindell et al., 2022])
  → application to images, point-clouds, 3D shapes, audio, video, etc.
- (∗) to **Multi-Task Learning** (Curriculum or Causal Learning [Wang et al., 2022])

Conclusion and Current Work

Take home messages for PINNs

- ✓ **Best of the data- and model-driven approach**
- ✓ Able to perform **unsupervised meshless evaluation** (vs. classical CNN)
- ✓ **Custom prior physical knowledge** (e.g., sub-grid models) as regularizers
- ✗ Some **hard constraint** can be implemented → ⚠ artifacts on HR and gradients
- ✗ Difficulty to minimize multi-task learning objectives (∗)

Links to other research fields

- to **Implicit Neural Representation** (**NeRF**, SIREN, BACON [Lindell et al., 2022])
  → application to images, point-clouds, 3D shapes, audio, video, etc.
- (∗) to **Multi-Task Learning** (Curriculum or Causal Learning [Wang et al., 2022])

Current work

- **Multi/Cross-scale training** with Fourier features at every layers
- Extension to temporal data and **PDEs**
- Need for **real data** (**do you have any?**)

Conclusion and Current Work

Take home messages for PINNs

- ✓ **Best of the data- and model-driven approach**
- ✓ Able to perform **unsupervised meshless evaluation** (vs. classical CNN)
- ✓ **Custom prior physical knowledge** (e.g., sub-grid models) as regularizers
- ✗ Some **hard constraint** can be implemented → ⚠ artifacts on HR and gradients
- ✗ Difficulty to minimize multi-task learning objectives (∗)

Links to other research fields

- to **Implicit Neural Representation** (**NeRF**, SIREN, BACON [Lindell et al., 2022]) → application to images, point-clouds, 3D shapes, audio, video, etc.
- (∗) to **Multi-Task Learning** (Curriculum or Causal Learning [Wang et al., 2022])

Current work

- **Multi/Cross-scale training** with Fourier features at every layers
- Extension to temporal data and **PDEs**
- Need for **real data** (**do you have any?**)

Thank you!

References i

Effinger, H. and Grossmann, S. (1987).
Static structure function of turbulent flow from the navier-stokes equations.
Zeitschrift für Physik B Condensed Matter, 66(3):289–304.

Lindell, D. B., Van Veen, D., Park, J. J., and Wetzstein, G. (2022).
Bacon: Band-limited coordinate networks for multiscale scene representation.
In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16252–16262.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2019).
Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations.
Journal of Computational Physics, 378:686–707.

Tancik, M., Srinivasan, P. P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J. T., and Ng, R. (2020).
Fourier features let networks learn high frequency functions in low dimensional domains.
NeurIPS.

Wang, S., Sankaran, S., and Perdikaris, P. (2022).
Respecting causality is all you need for training physics-informed neural networks.
arXiv preprint arXiv:2203.07404.