Introduction

ADVANCED DEEP LEARNING FOR PHYSICS, COURSE OVERVIEW

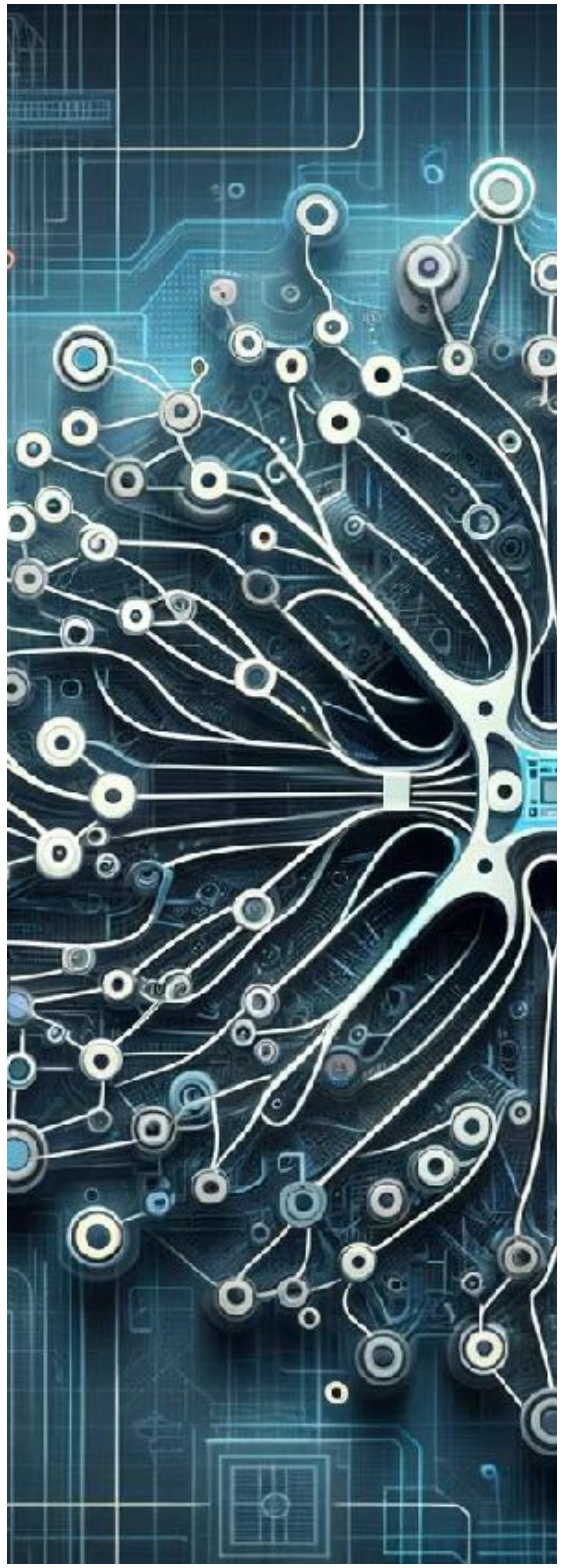
General Motivation

Computational Methods in the Age of Deep Learning

- Traditional simulation methods have shown tremendous success
  - Magnetohydrodynamics
  - Turbo machinery and combustion
  - Aerodynamic shape optimisation
  - Blood flow in a human heart
  - …

General Motivation

Computational Methods in the Age of Deep Learning



- Traditional simulation methods have shown tremendous success
- A.I. / "*Deep Learning*" comes along.
  - Can find cats & dogs in images
  - Alpha Go
  - Alphafold 1,2
  - Chat GPT
- **Open question: *how to join forces of both worlds***

Connections to Neuroscience and Psychology

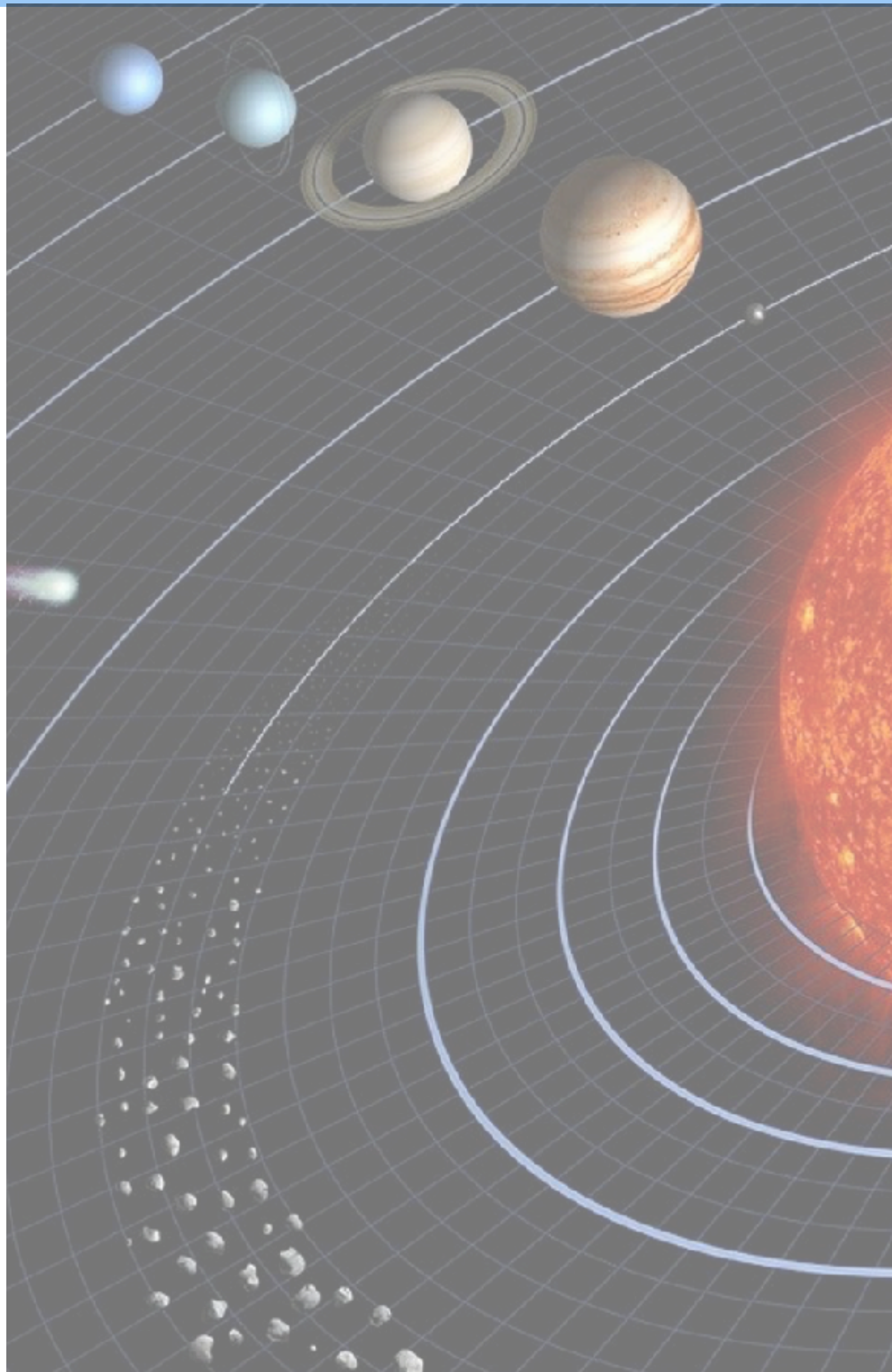
Thinking fast and slow (D. Kahnemann)

- System 1 - fast, intuition
- System 2 - analytic thinking
- Goal:
  - Simulations provide *“system 2”* functionality
  - Combine with improved intuition via *“system 1”* from NNs

Machine Learning in Science

Scientific Discoveries

- Traditional: experiment & theory. Contemporary: computation.
- Machine learning & data-driven approaches:
  - Understanding nature: Keppler's orbit (1609)
  - Discovering patterns from observations (Darcy's law, 1856)
  - Principal component analysis (e.g. Pearson, 1901)
- Deep Learning as fundamental step forward



Versus Classical Numerical Methods

Advancements in scientific disciplines

- Case 1) Brilliant insight / flash of light / theory
  - Einstein: theory of general relativity
  - Many derivative works follow…
- Case 2) Long, arduous path, tiny steps, many unsolved questions
  - Navier-Stokes equations: (Millennium prize problem) still unproved
  - Hard work, chipping away to uncover the nature of things…

Versus Classical Numerical Methods

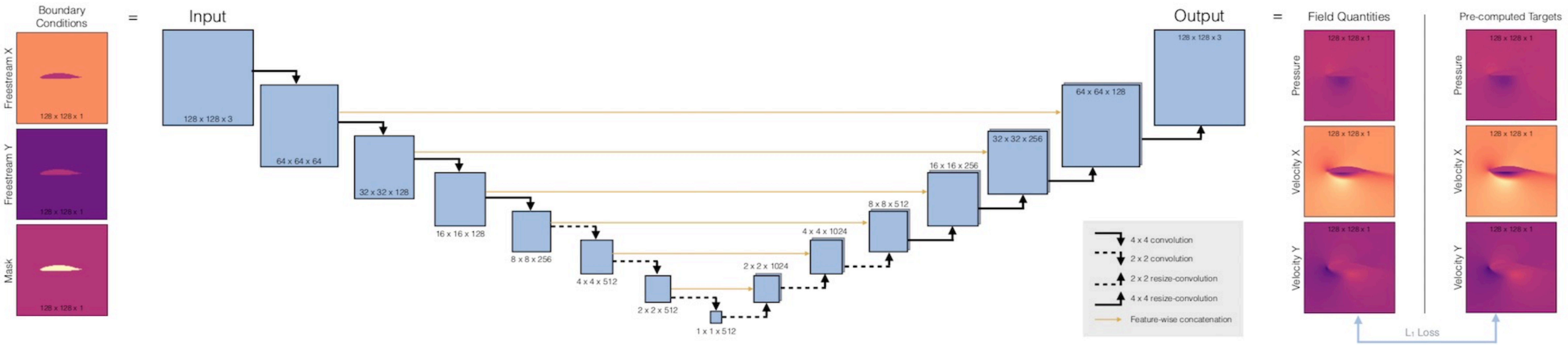
- *Traditional numerical methods* had a difficult start (cf. Case 2)
    - Initially frowned upon
    - “Unreliable” , “inexact”, etc. …
    - Now we know: extremely useful tools
- Deep Learning also clearly of *“the second kind”*
- No reason *not* to look into it

Rough Categorization - Types of Integration

Category 1/3: Supervised / No Integration

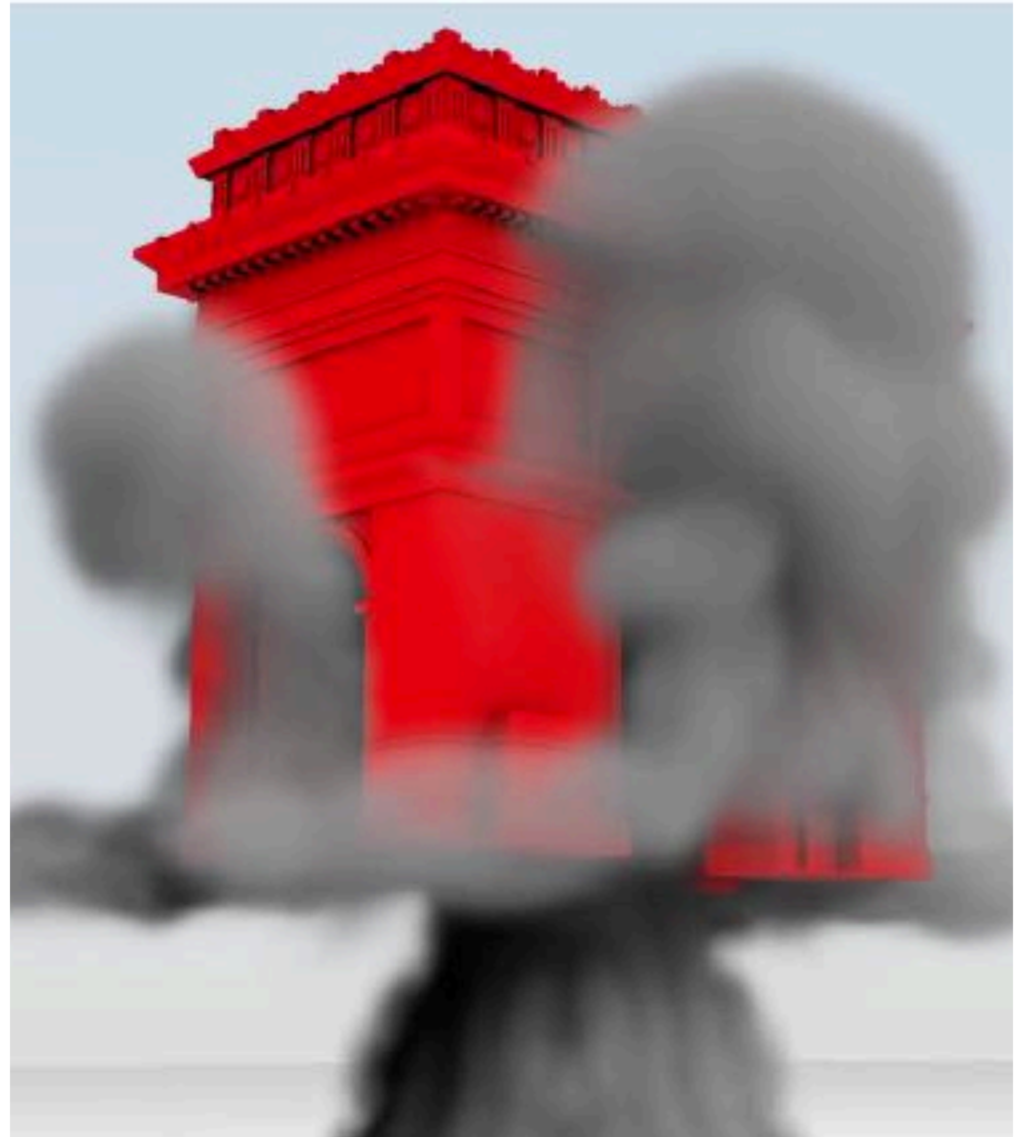
E.g.: https://github.com/thunil/Deep-Flow-Prediction

- Supervised loss, differential equation (PDE / ODE) only used to generate data
- Thuerey et. al: learning RANS flows



Category 2/3: Loss Terms / “Unsupervised”

E.g.: https://github.com/google/FluidNet

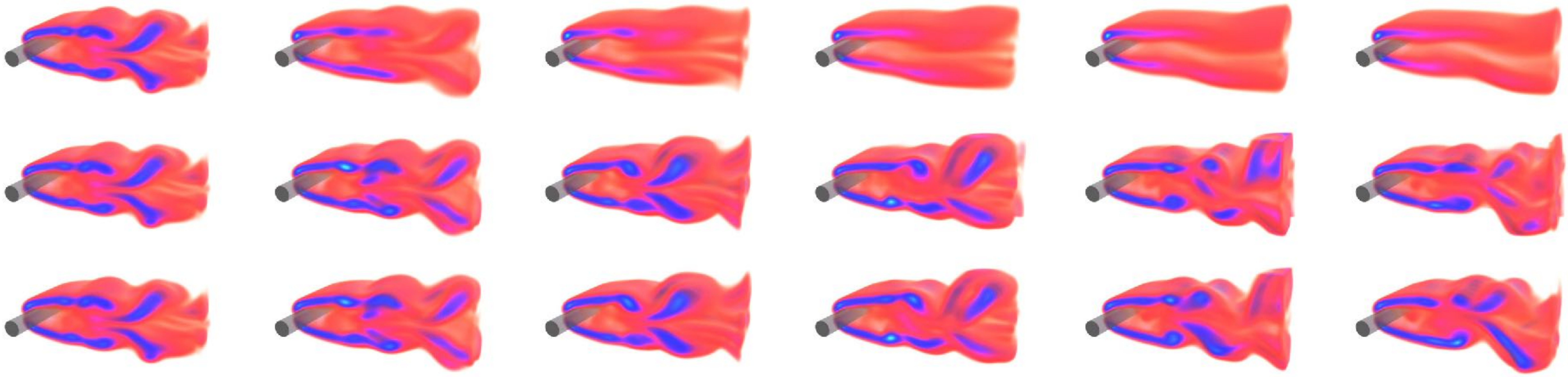


- “Unsupervised” training
- Tompson et al.: learning Poisson solvers
- In a nutshell: minimize $\nabla \cdot f(x)$
- Raissi et al.: “physics-informed” networks, PINNs
- Physics objective: conservation of mass via divergence free velocity field

Category 3/3: Hybrid / Tight Integration

E.g.: https://github.com/tum-pbs/Solver-in-the-Loop

- Train neural network to work alongside numerical solver *(details will follow)*
- Um et al.: *“solver in the loop”* approach
- Unsuper-/supervised distinction not too meaningful



Hybrid Solver Example

Example in Motion: Unsteady Wake Flow in 3D, Re=546.9

Still using
“source” solver!

Content

1. Introduction
2. Supervised Learning & Architectures
3. Physical Losses & Differentiable Solvers
4. Learning Tasks with Differentiable Physics
5. Graph Networks
6. Time-Series Predictions
7. Diffusion Models
8. Reinforcement Learning
9. Conclusions

Content

What this course is _not_

- No introduction to deep learning
  - Expected to know: GD, backprop, etc. …
  - Self-check: difference between Autoencoder, U-net, and ResNet?
- No introduction to numerical simulation
  - Expected to know: finite differences & co., basic iterative solvers
- [But: both topics not super complicated, can potentially be learned on the side]
- Focus lies on methods on the intersection of DL & simulations

Script

Primarily: https://www.physicsbaseddeeplearning.org/

- Plus PDF slides
- No separate script…

End