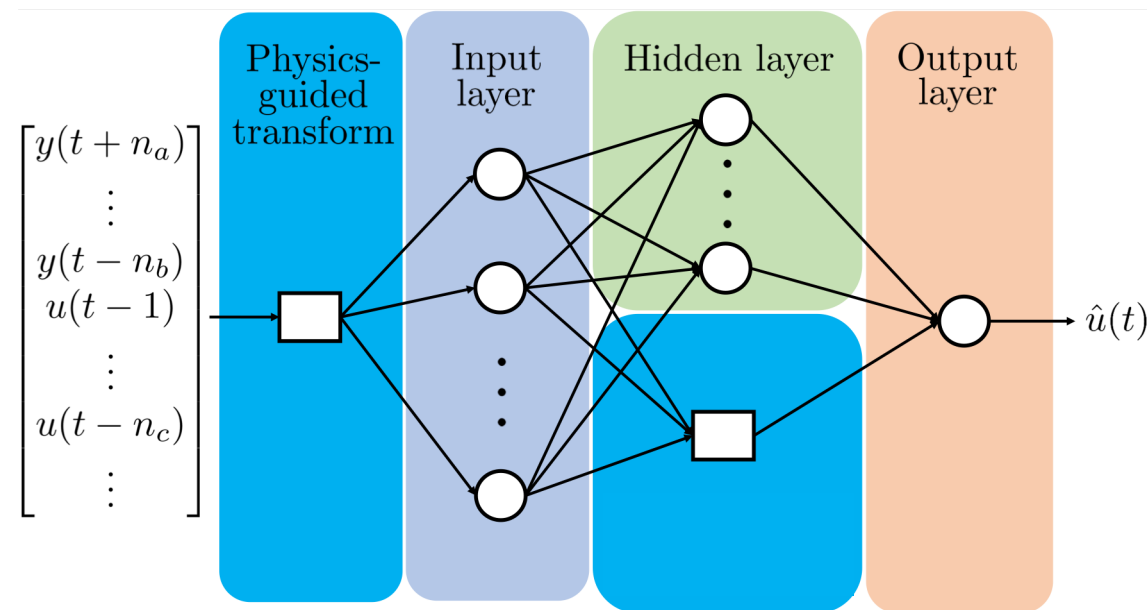
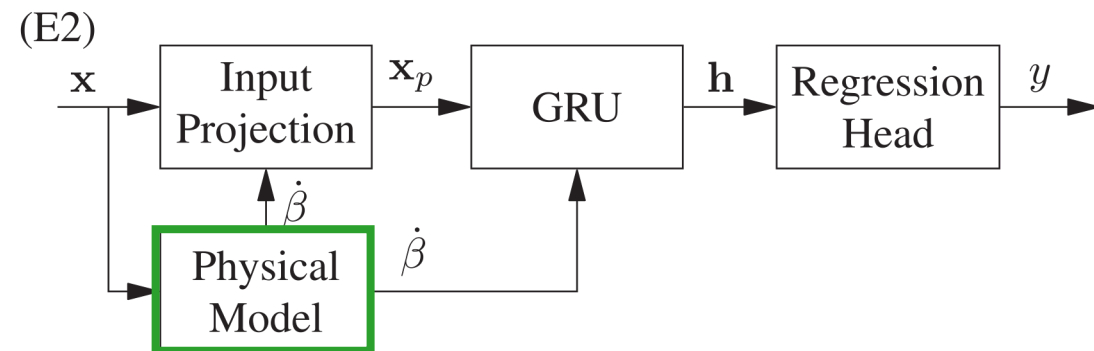


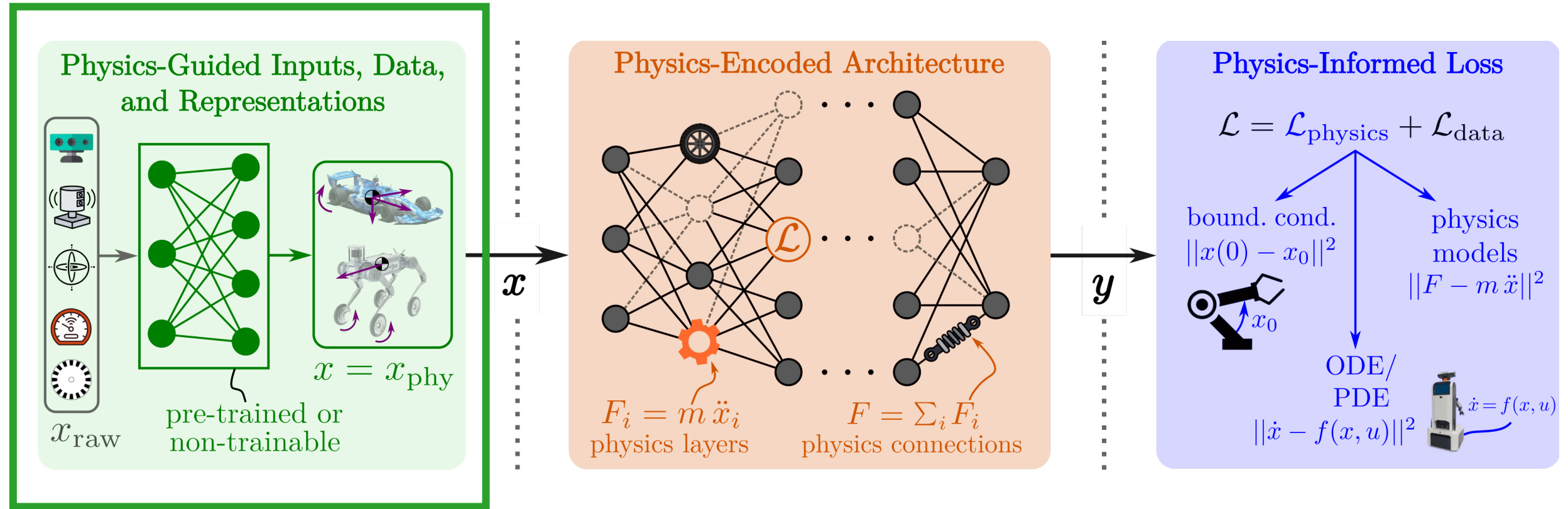
Physics-Guided Inputs, Data and Representations

Dr. Mattia Piccinini &
Prof. Gastone Pietro Rosati Papini

University of Trento &
Technical University of Munich,
Autonomous Vehicle Systems Lab



Physics-Guided Learning: Positioning



What Does Physics-Guided Learning Include?

Embedding **physics insights** into:

- Input features
- Data representations
- Training data selection / generation



ASME
SETTING THE STANDARD

ASME Journal of Computing and Information Science in Engineering
Online journal at:
<https://asmedigitalcollection.asme.org/computingengineering>



Salah A. Faroughi¹
Geo-Intelligence Laboratory,
Ingram School of Engineering,
Texas State University,
San Marcos, TX 78666
e-mail: salah.faroughi@txstate.edu

Nikhil M. Pawar
Geo-Intelligence Laboratory,
Ingram School of Engineering,
Texas State University,
San Marcos, TX 78666

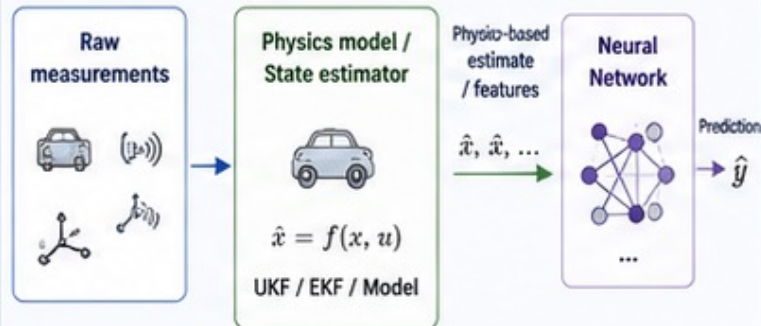
Célio Fernandes
Geo-Intelligence Laboratory,

Physics-Guided Physics-Informed, and Physics-Encoded Neural Networks and Operators in Scientific Computing: Fluid and Solid Mechanics

Physics-Guided Learning: Multiple Classes

1 PHYSICS MODELS AS INPUTS FOR NEURAL NETWORKS

Use outputs of physics-based models or state estimators as structured inputs to neural networks.



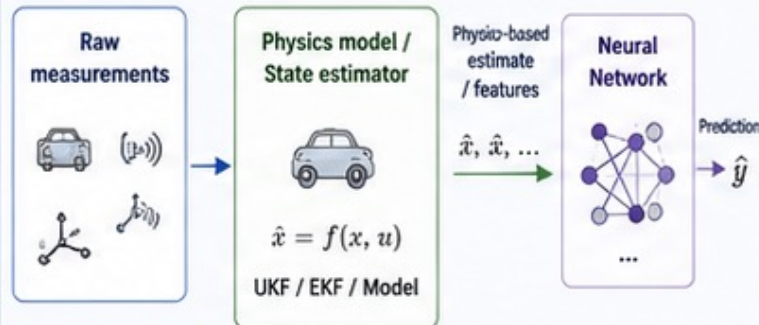
Examples

- Vehicle sideslip angle estimation (Graber et al., 2018)
- Slider-crank acceleration prediction (De Groote et al., 2021)
- Lateral velocity estimation with UKF + NN (Cortese & Lenzo, 2026)
- Quadrotor dynamics learning in MPC (Hong et al., 2023)
- Physics-guided inputs for motor control (Bolderman et al., 2021)

Physics-Guided Learning: Multiple Classes

1 PHYSICS MODELS AS INPUTS FOR NEURAL NETWORKS

Use outputs of physics-based models or state estimators as structured inputs to neural networks.



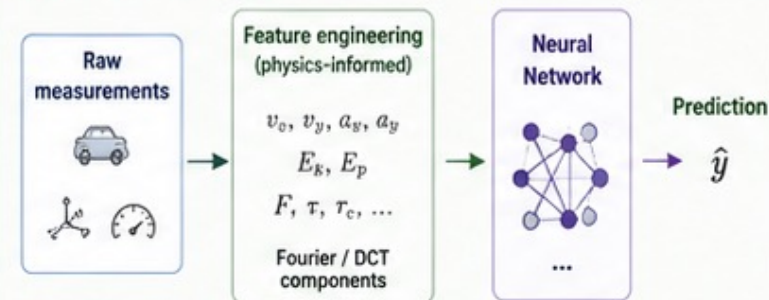
Examples

- Vehicle sideslip angle estimation (Graber et al., 2018)
- Slider-crank acceleration prediction (De Groote et al., 2021)
- Lateral velocity estimation with UKF + NN (Cortese & Lenzo, 2026)
- Quadrotor dynamics learning in MPC (Hong et al., 2023)
- Physics-guided inputs for motor control (Bolderman et al., 2021)

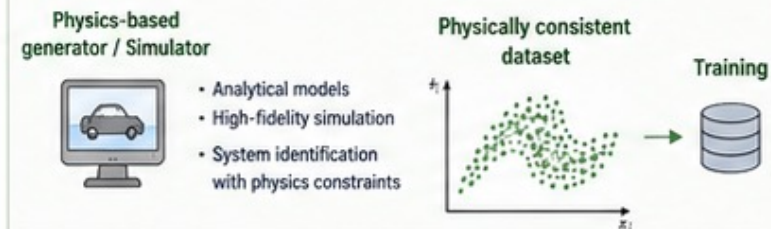
2 PHYSICS-GUIDED FEATURES AND TRAINING DATA

Design physics-informed input features and generate or select training data that obey physical laws.

A) Physics-guided features



B) Physically consistent training data



Examples

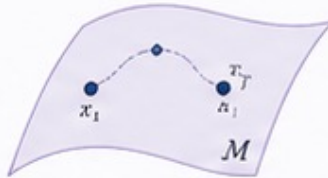
- Physics-guided features for manipulators (De Groote et al., 2021)
- Fourier/DCT features for locomotion (Wu et al., 2022; O'Connor et al., 2025)
- Data generation with physical simulators (Li et al., 2025; Mao et al., 2025)
- Differentiable rendering & augmentation (Li et al., 2025; Shang et al., 2026)

Physics-Guided Learning: Multiple Classes

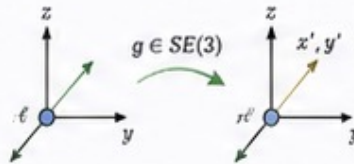
3 GEOMETRIC LEARNING

Respect geometric structures (manifolds, symmetry, SE(3) transforms, graphs) in data and models.

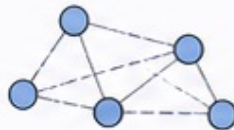
Manifold / Lie group learning



SE(3) equivariant models



Graph / Interaction learning



Geometric constraints in latent space



Examples

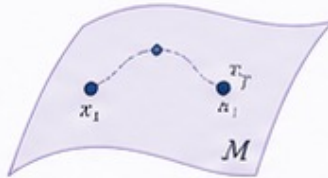
- Riemannian flow matching on SE(3) (Chen & Lipman, 2023)
- Equivariant GNNs for traffic (Chen et al., 2024)
- SE(3) equivariant dynamics (Shah et al., 2016; Falco et al., 2019)
- Differentiable kinematics for manipulation (Lv et al., 2025)
- SE(3)-equivariant diffusion policies (Tucker et al., 2026)

Physics-Guided Learning: Multiple Classes

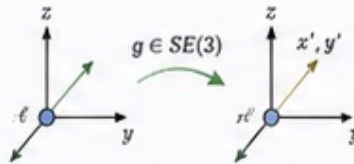
3 GEOMETRIC LEARNING

Respect geometric structures (manifolds, symmetry, SE(3) transforms, graphs) in data and models.

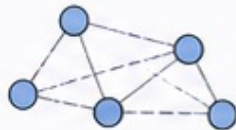
Manifold / Lie group learning



SE(3) equivariant models



Graph / Interaction learning



Geometric constraints in latent space



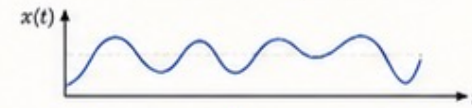
Examples

- Riemannian flow matching on SE(3) (Chen & Lipman, 2023)
- Equivariant GNNs for traffic (Chen et al., 2024)
- SE(3) equivariant dynamics (Shah et al., 2016; Falco et al., 2019)
- Differentiable kinematics for manipulation (Lv et al., 2025)
- SE(3)-equivariant diffusion policies (Tucker et al., 2026)

4 FREQUENCY-DOMAIN LEARNING

Transform data to the frequency domain to capture dynamics across time and space scales.

Time-domain signals



Transform
(DCT / FFT)

Frequency-domain representation



Modeling in frequency domain



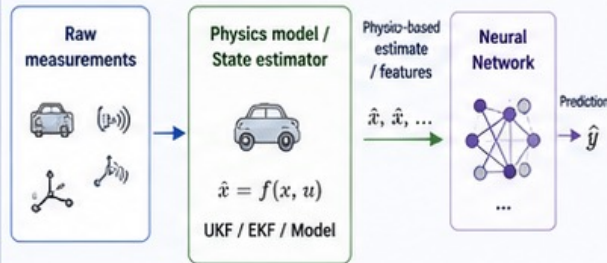
Examples

- Spatiotemporal dynamics with DCT (Wu et al., 2022; Wu et al., 2025)
- Low-frequency components for hexapod gaits (O'Connor et al., 2025)
- Fourier features for human motion (Shah et al., 2016; Falco et al., 2019)

Physics-Guided Learning: Multiple Classes

1 PHYSICS MODELS AS INPUTS FOR NEURAL NETWORKS

Use outputs of physics-based models or state estimators as structured inputs to neural networks.



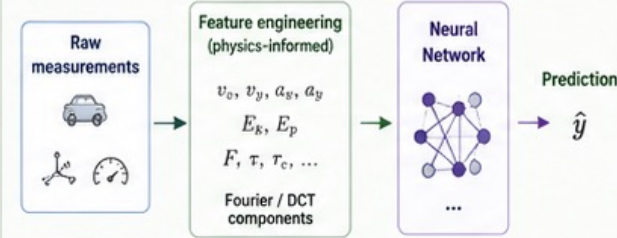
Examples

- Vehicle sideslip angle estimation (Graber et al., 2018)
- Slider-crank acceleration prediction (De Groote et al., 2021)
- Lateral velocity estimation with UKF + NN (Cortese & Lenzo, 2026)
- Quadrotor dynamics learning in MPC (Hong et al., 2023)
- Physics-guided inputs for motor control (Bolderman et al., 2021)

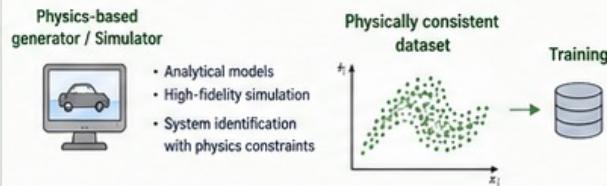
2 PHYSICS-GUIDED FEATURES AND TRAINING DATA

Design physics-informed input features and generate or select training data that obey physical laws.

A) Physics-guided features



B) Physically consistent training data



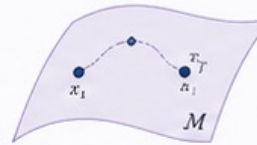
Examples

- Physics-guided features for manipulators (De Groote et al., 2021)
- Fourier/DCT features for locomotion (Wu et al., 2022; O'Connor et al., 2025)
- Data generation with physical simulators (Li et al., 2025; Mao et al., 2025)
- Differentiable rendering & augmentation (Li et al., 2025; Shang et al., 2026)

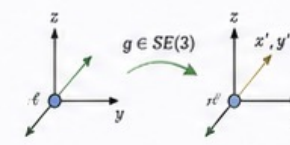
3 GEOMETRIC LEARNING

Respect geometric structures (manifolds, symmetry, SE(3) transforms, graphs) in data and models.

Manifold / Lie group learning



SE(3) equivariant models



Graph / Interaction learning



Geometric constraints in latent space

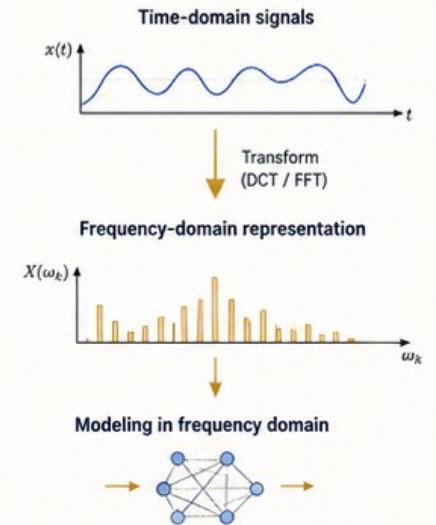


Examples

- Riemannian flow matching on SE(3) (Chen & Lipman, 2023)
- Equivariant GNNs for traffic (Chen et al., 2024)
- SE(3) equivariant dynamics (Shah et al., 2016; Falco et al., 2019)
- Differentiable kinematics for manipulation (Lv et al., 2025)
- SE(3)-equivariant diffusion policies (Tucker et al., 2026)

4 FREQUENCY-DOMAIN LEARNING

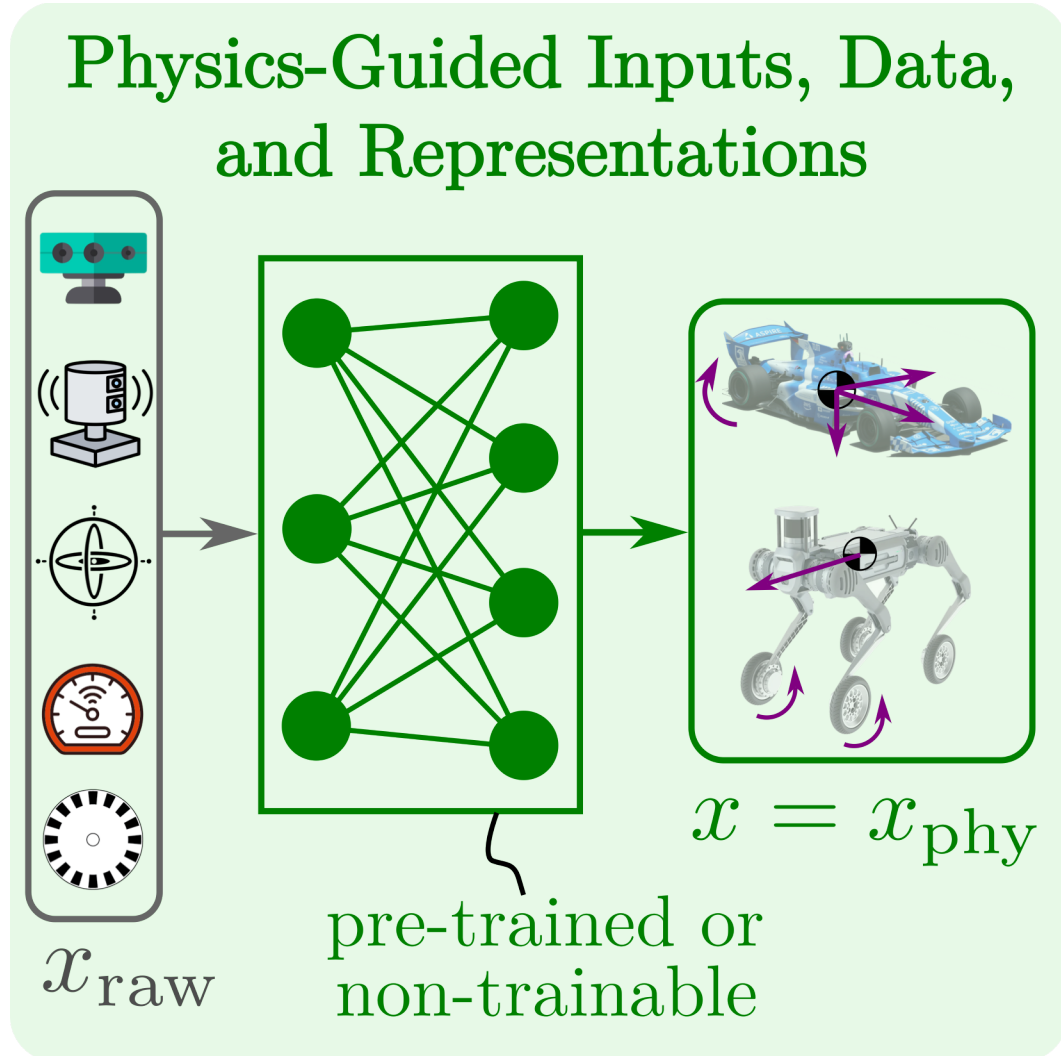
Transform data to the frequency domain to capture dynamics across time and space scales.



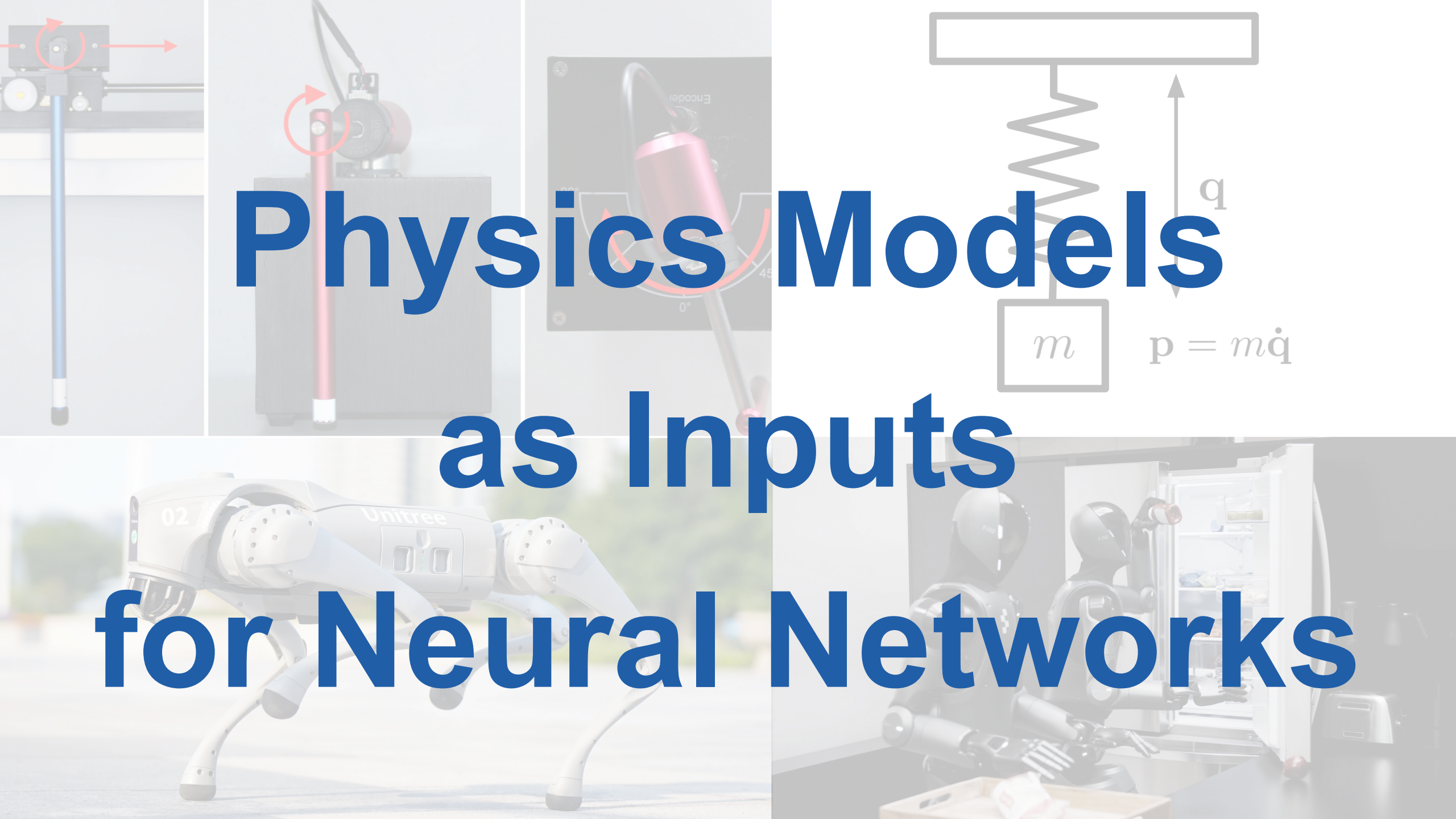
Examples

- Spatiotemporal dynamics with DCT (Wu et al., 2022; Wu et al., 2025)
- Low-frequency components for hexapod gaits (O'Connor et al., 2025)
- Fourier features for human motion (Shah et al., 2016; Falco et al., 2019)

Physics-Guided Learning: Remark



Physics-guided operations are **pre-trained** or **non-trainable**:
if they contain NNs, these are ***not* trained** together with the main model

The background is a collage of four images. Top-left: A close-up of a mechanical joint with a red arrow indicating rotation. Top-right: A diagram of a mass-spring system with a mass 'm' and displacement 'q', with the equation 'p = m\dot{q}' below it. Bottom-left: A white quadruped robot, labeled 'Unitree' and '02', in a running pose. Bottom-right: A robotic arm in a kitchen setting, reaching into an open refrigerator.

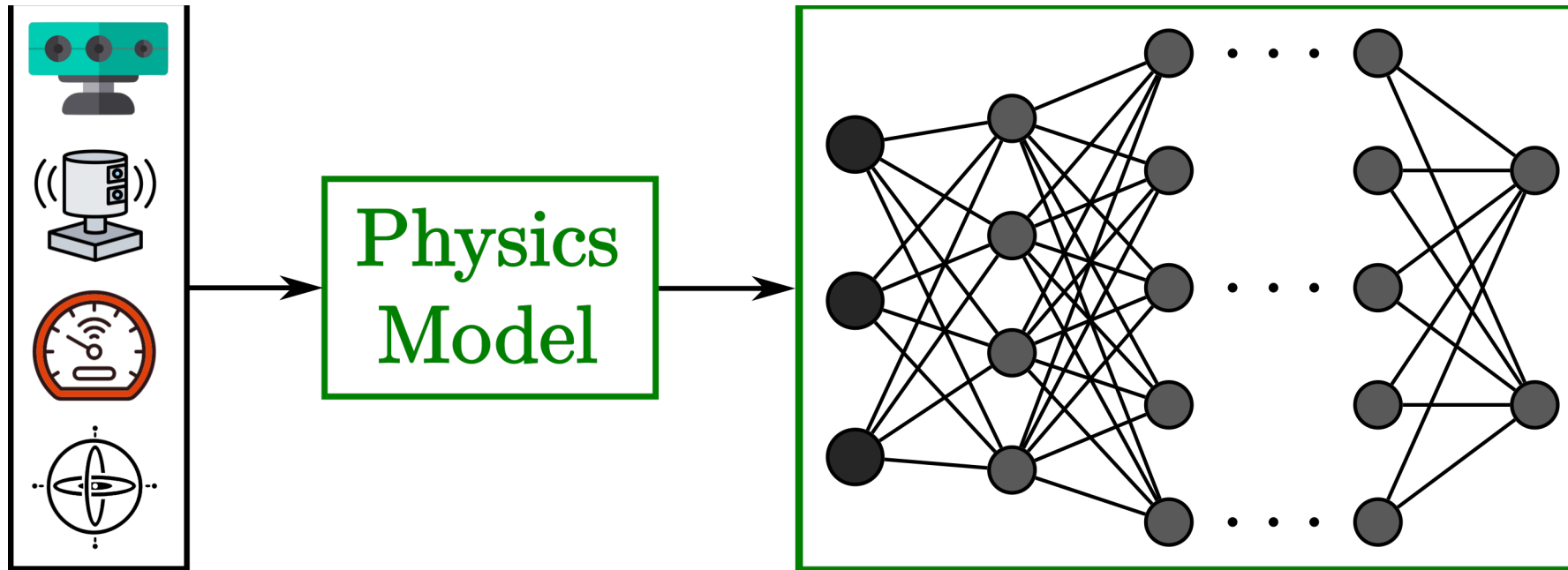
Physics Models

as Inputs

for Neural Networks

Physics Models as Inputs for Neural Networks

Idea: Use **physics-based modes** to generate input features for machine learning models
→ Constrain learning to a **physically consistent** space

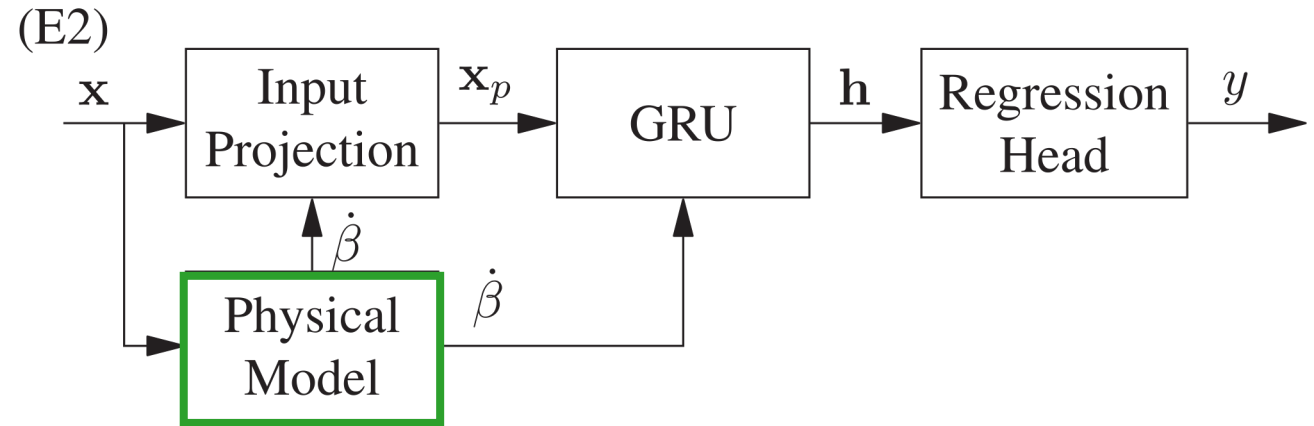
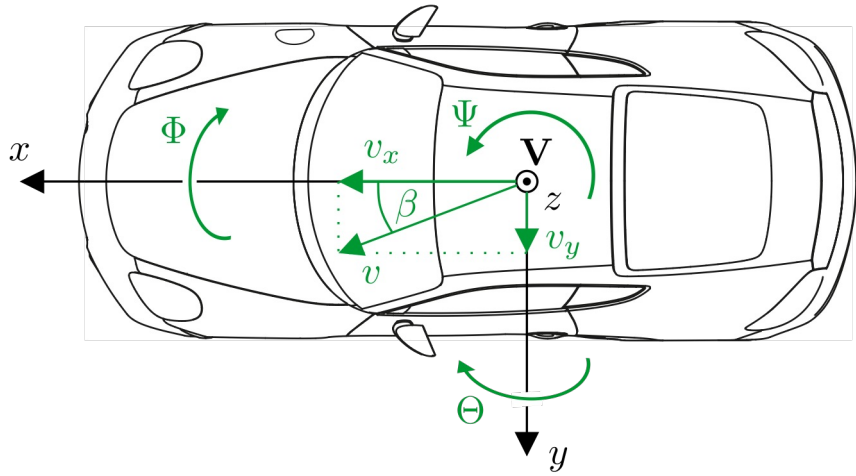


Physics Models as Inputs for Neural Networks - Examples

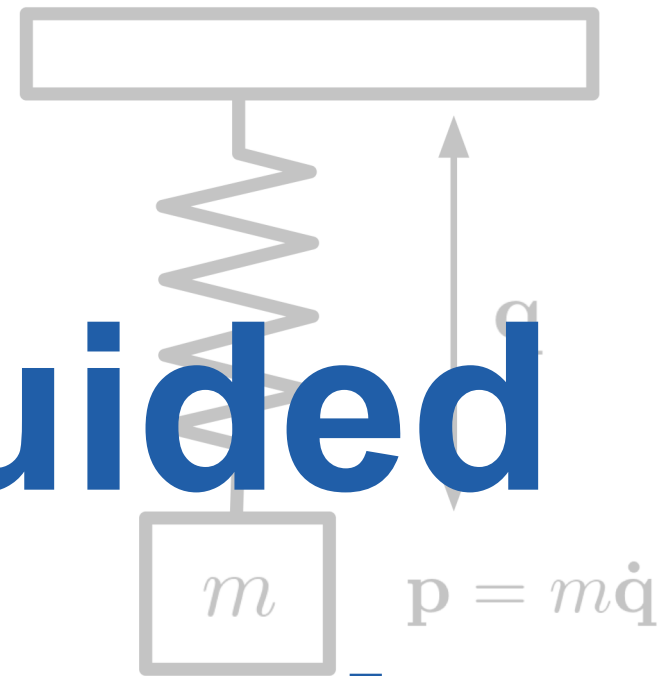
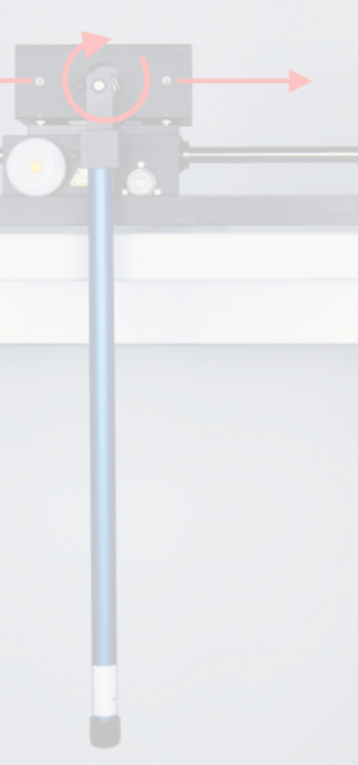
Vehicle sideslip angle (β) estimation

Physics-guided NN design:

- **Physics-based estimator** of $\dot{\beta}$, then fed to a Gated Recurrent Unit (**GRU**) NN
- **Outperforming** the same NN without physics-guided inputs
- **Outperforming** a purely physics-based model



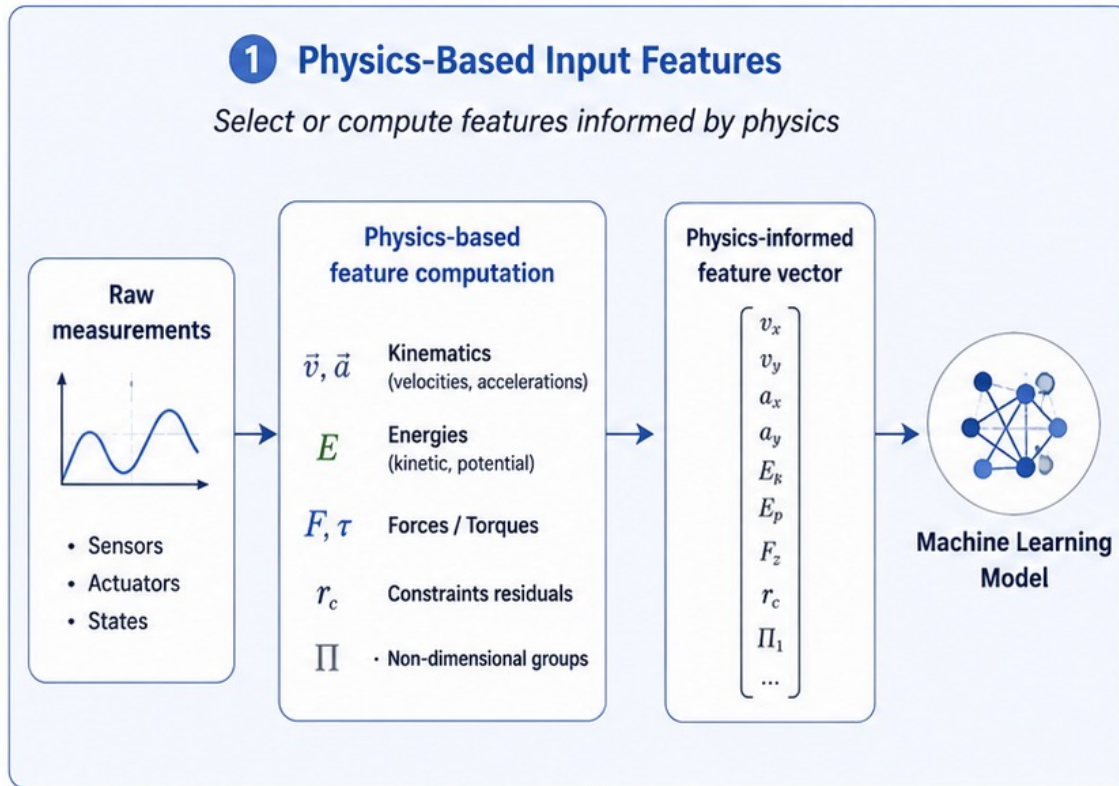
Physics-Guided Features and Training Data



Physics-Guided Features and Training Data

Idea:

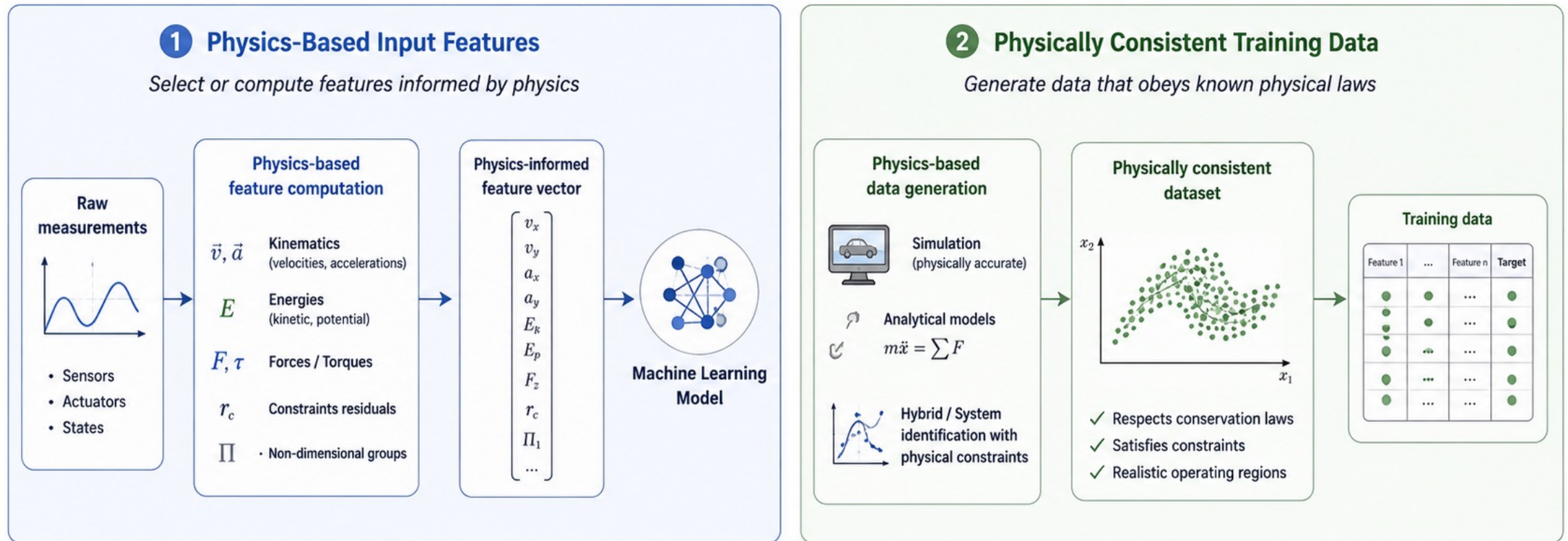
- Select / compute **physics-based input features** for machine learning models



Physics-Guided Features and Training Data

Idea:

- Select / compute **physics-based input features** for machine learning models
- Generate physically consistent **training data**

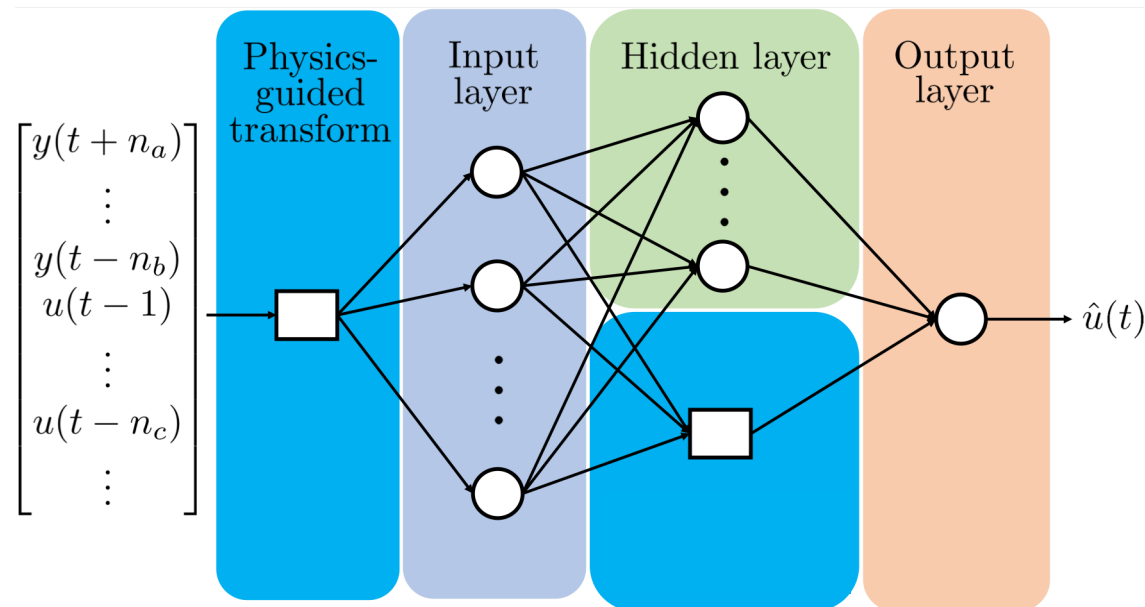


Physics-Guided Features and Training Data - Examples

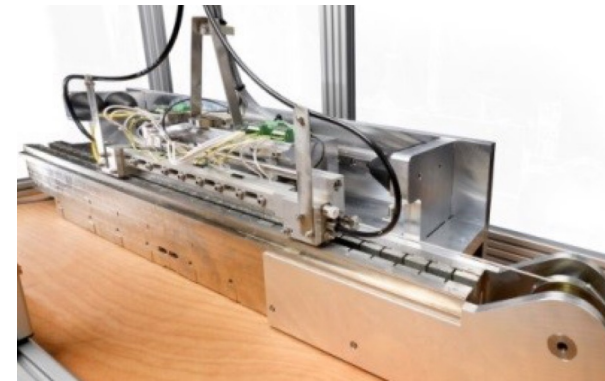
Feedforward control of linear motors: Compute force $\hat{u}(t)$ to track a desired position y

Physics-guided NN design:

- Linear friction depends on position y , velocity $\dot{y}$, $\text{sign}(\dot{y}) \rightarrow$ use them as **additional NN inputs**
- **Outperforming** the same NN without physics-guided input features



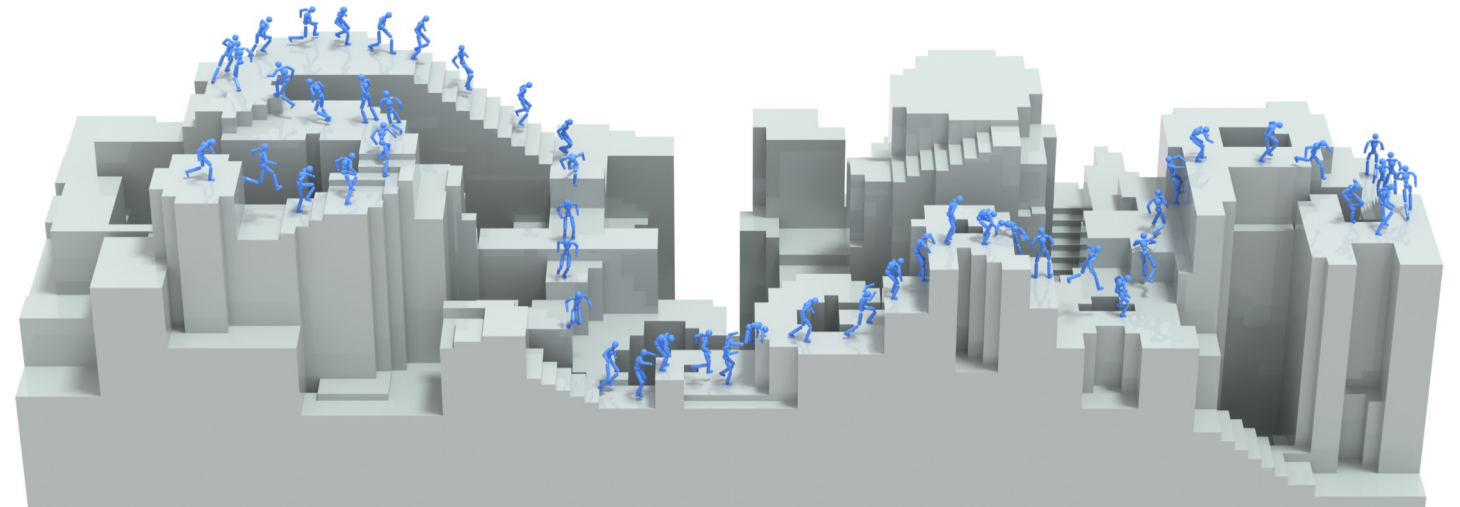
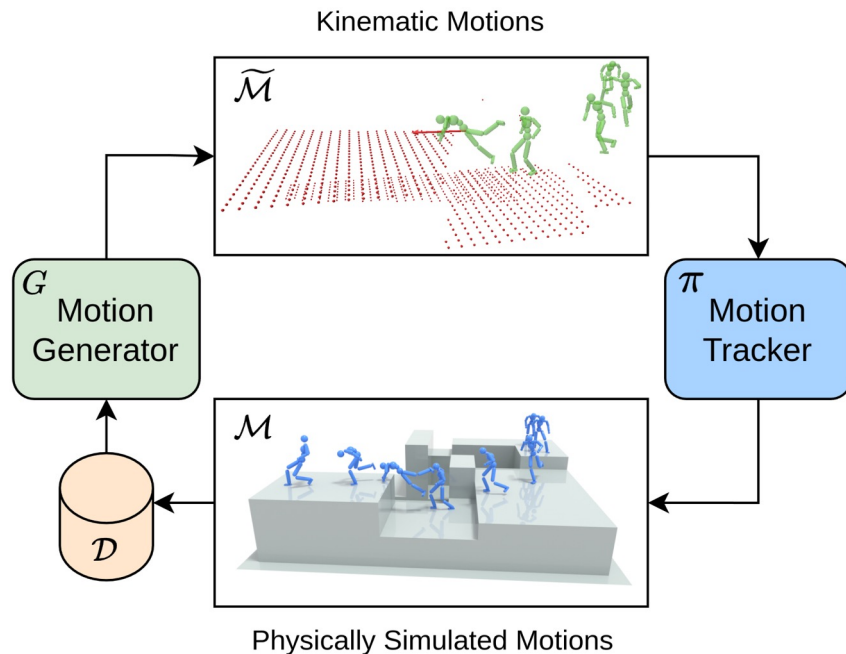
$$\hat{u}(t) = f_{\text{PGNN-I}}\left(\theta, \ddot{y}(t), u(t-1), y(t), y(t-1), \dot{y}(t), \dot{y}(t-1), \text{sign}(\dot{y}(t)), \text{sign}(\dot{y}(t-1))\right).$$



Physics-Guided Features and Training Data - Examples

Generating human motions in **parkour** scenarios in Isaac Gym

- **Diffusion model** to generate motions
- **Physics-based tracking controller** to make the motions physically consistent
- The corrected motions are recycled as **training data** for the **diffusion model**



Questions & Discussion

