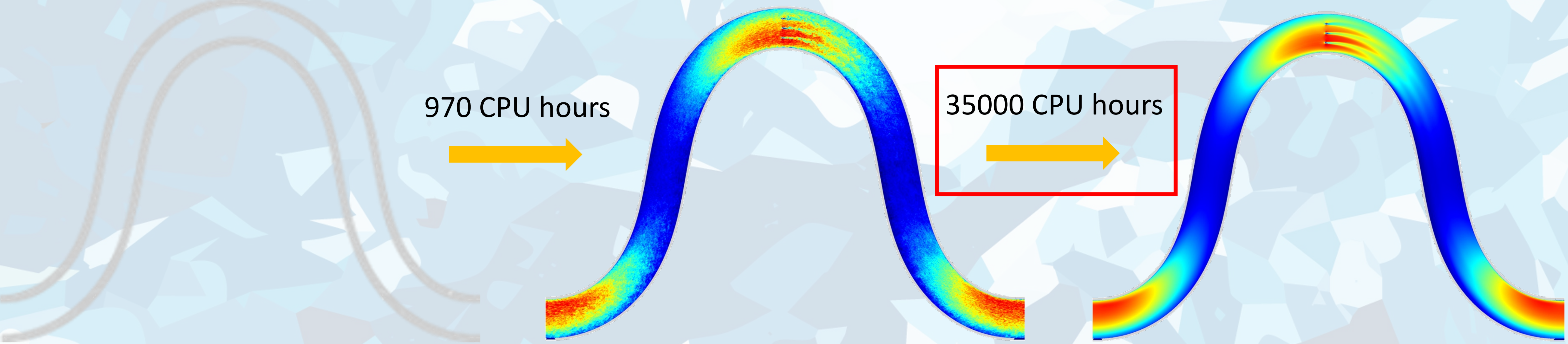
Turbulent flow predictions of MHK turbine arrays using physics-informed convolutional neural networks

Zexia Zhang and Ali Khosronejad
Civil Engineering Department, College of Engineering & Applied Sciences,
Stony Brook University

Motivation

Large Eddy Simulations (LES) of large-scale rivers are very expensive, especially on calculating statistical properties, i.e. $\bar{u}$, $\bar{v}$, $\bar{w}$, $\overline{u'u'}$, $\overline{v'v'}$, $\overline{w'w'}$, etc.



Large- scale virtual meandering river

River size: 2110 m × 100 m × 3.3 m

Grid: 6601 × 501 × 21 = 6.9 × 10^7

Fully-converged instantaneous turbulent flow field

Time-averaged flow field

Motivation

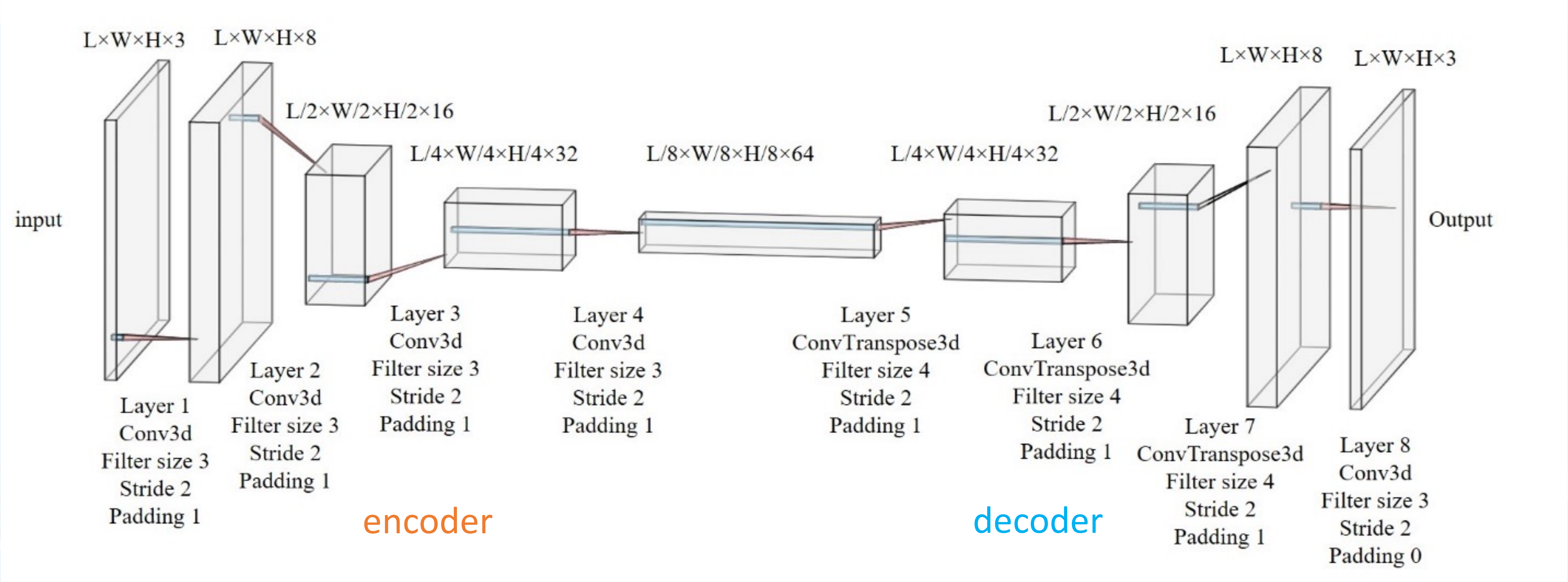
Can we use artificial intelligence (AI) techniques to reduce the cost?

Strategy

Fully-converged instantaneous turbulent flow field

Time-averaged flow field

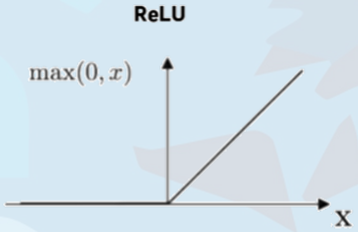
Convolutional Neural Network (CNN)



Output Target

update kernel and bias

loss



$$Layer_{out} = \sigma(Layer_{in} \otimes kernel + bias)$$

σ: activation function

Loss function and Physics constraints

Governing equations for mean velocity:

$$Div = 0$$
$$M = Conv + P + Diff + RS$$

Difference between LES and CNN results:

$$\Delta Div = Div_{CNN} - Div_{LES}$$
$$\Delta M = M_{CNN} - M_{LES}$$

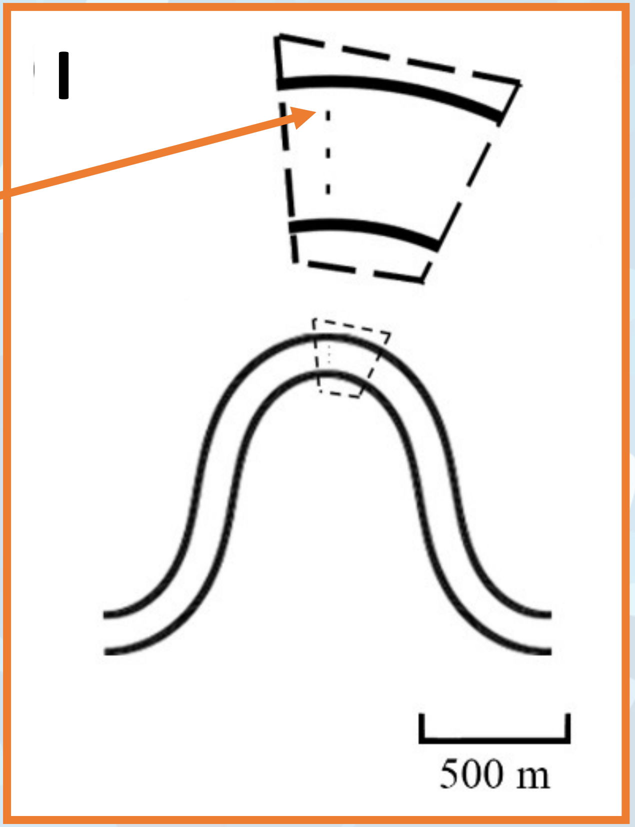
Loss function of CNN:

CNN **PICNN**

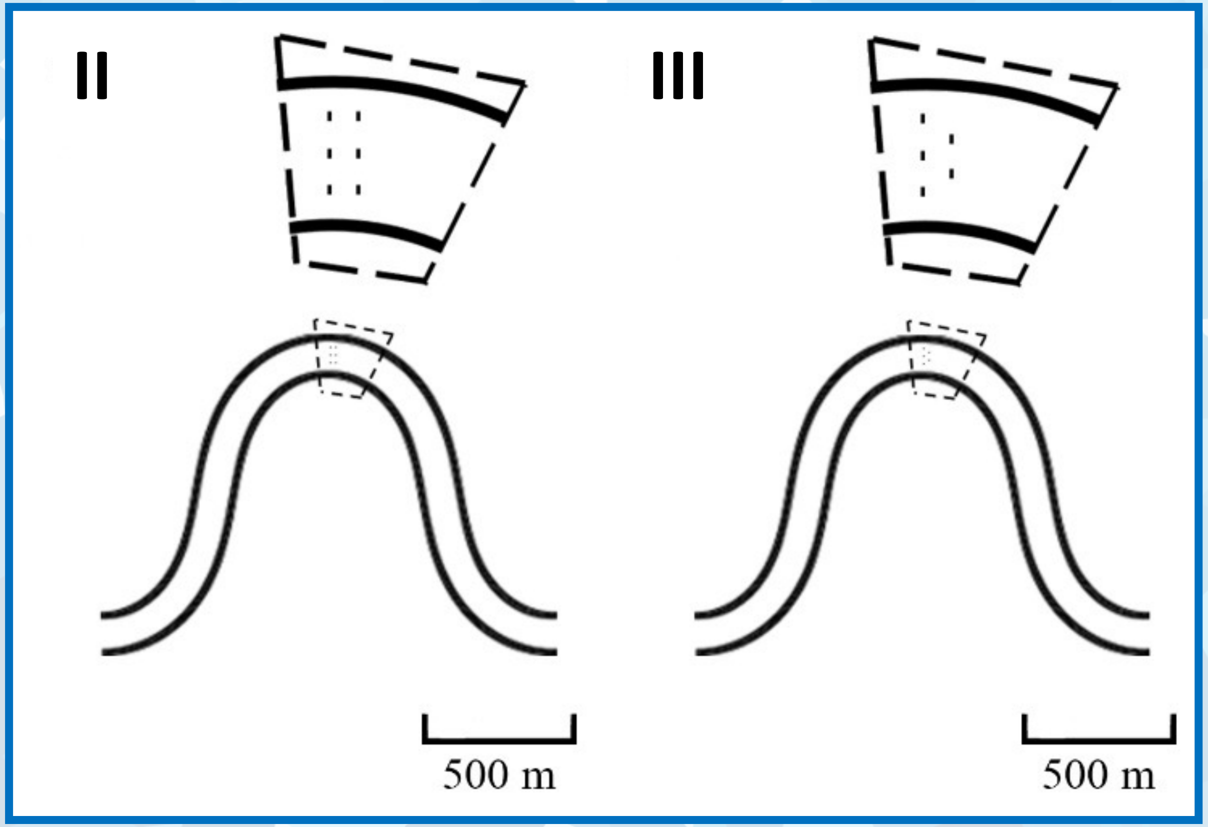
$$Loss = MSE(\Phi_{CNN} - \Phi_{LES}) + MSE(\Delta Div) + MSE(\Delta M)$$

Training and validation cases

Turbines
D = 1.5 m



Training case



Validation cases

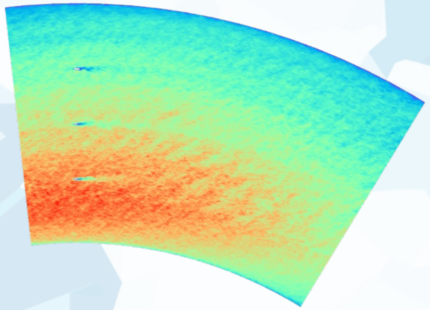
* Virtual Rivers are developed by Ajay B. Limaye from Department of Environmental Sciences, University of Virginia

Training- mean velocity

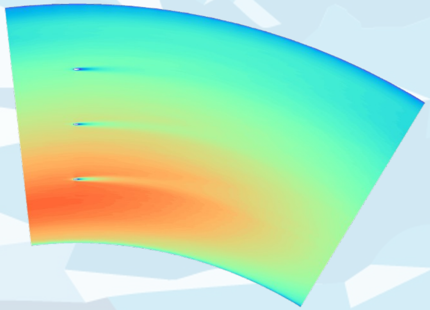
- Run LES to produce training data
  - Instantaneous velocity fields
  - Time-averaged velocity fields

- Train the AI model
  - Learning rate: 0.001
  - Optimizer: Adam
  - Converge Epochs: 1400

Training sample



Inputs: instantaneous 3D flow fields

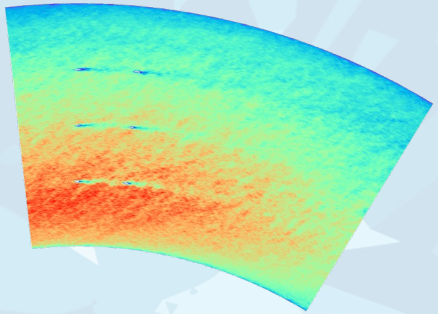


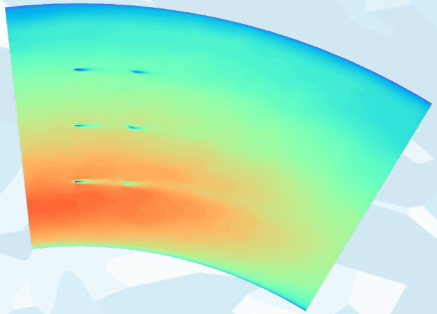
Training target: time-averaged 3D flow field

77 samples

Training dataset

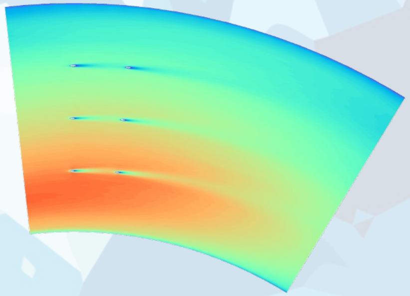
Validation- mean velocity





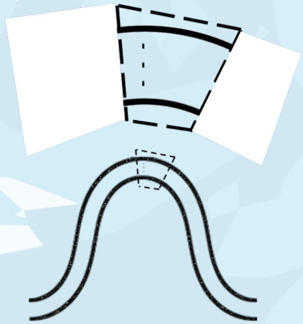
LES → Instantaneous flow field → AI → Prediction of time-averaged flow field

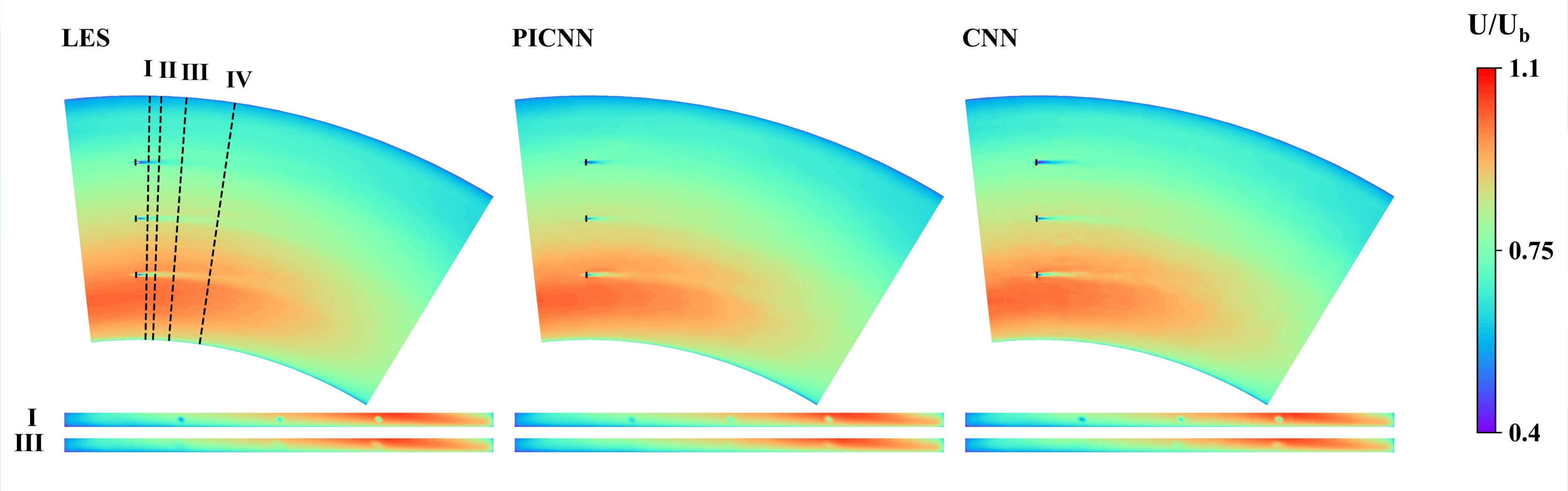
Instantaneous flow field → Time-averaged LES Result

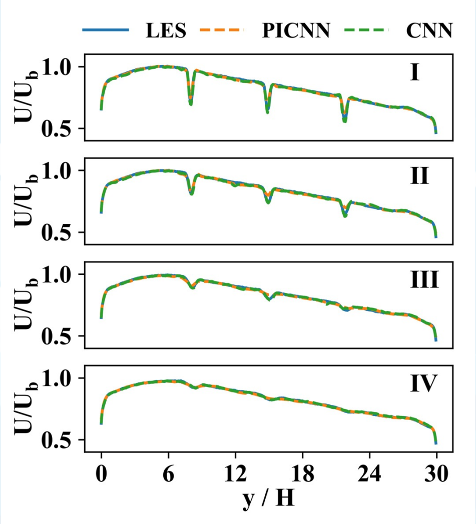


Results- training case

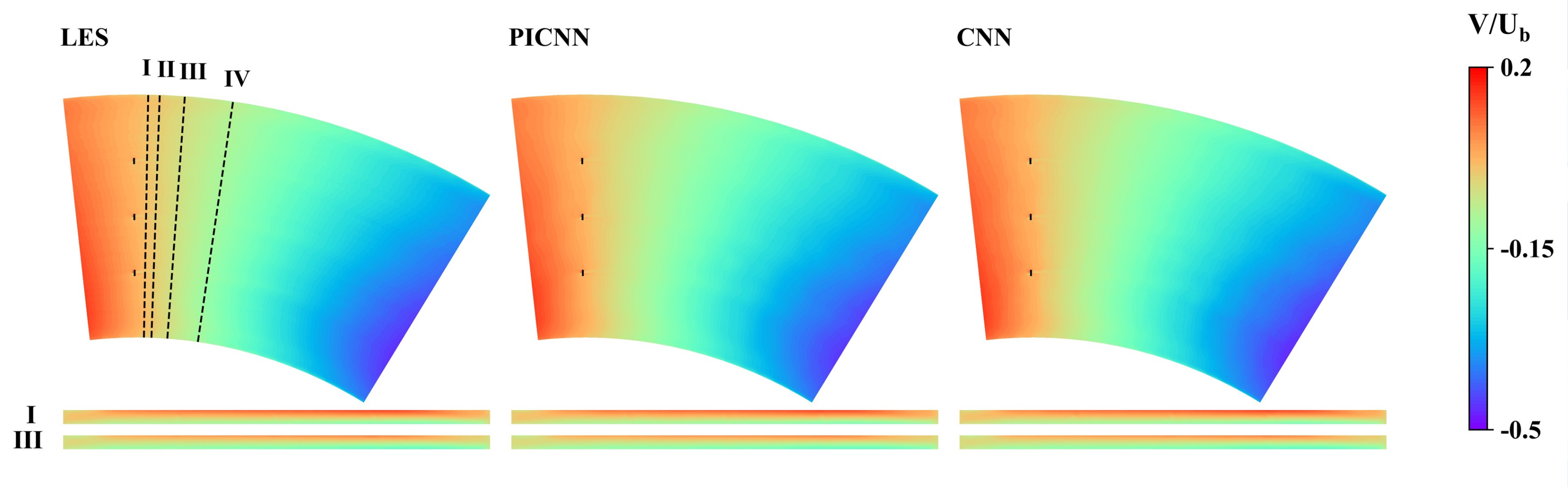
$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$

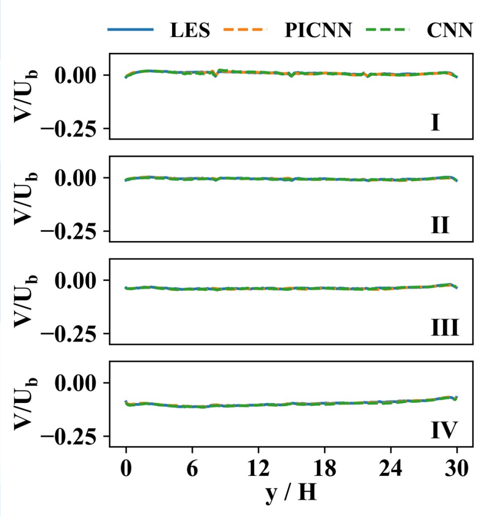






Percentage error:
PICNN: 0.57%
CNN:0.57%

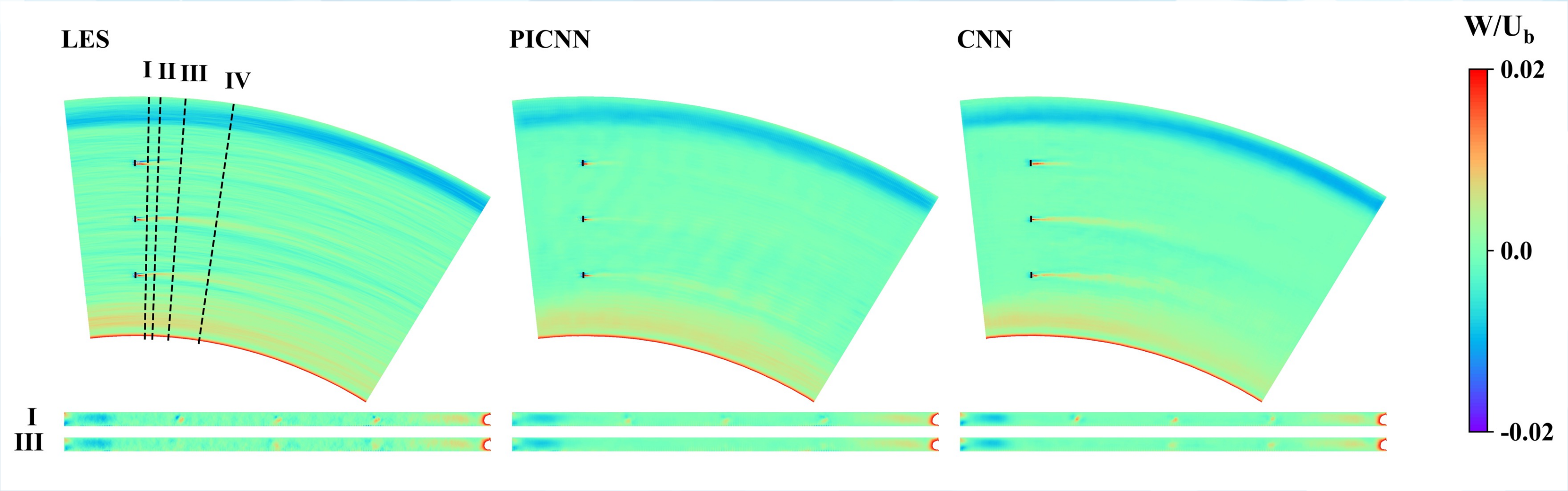


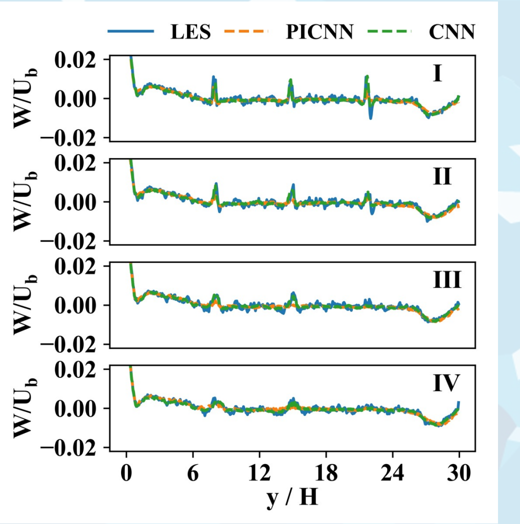


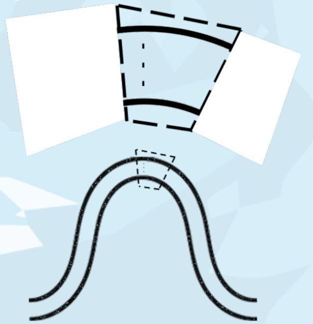
Percentage error:
PICNN: 1.23%
CNN: 1.79%

Results- training case

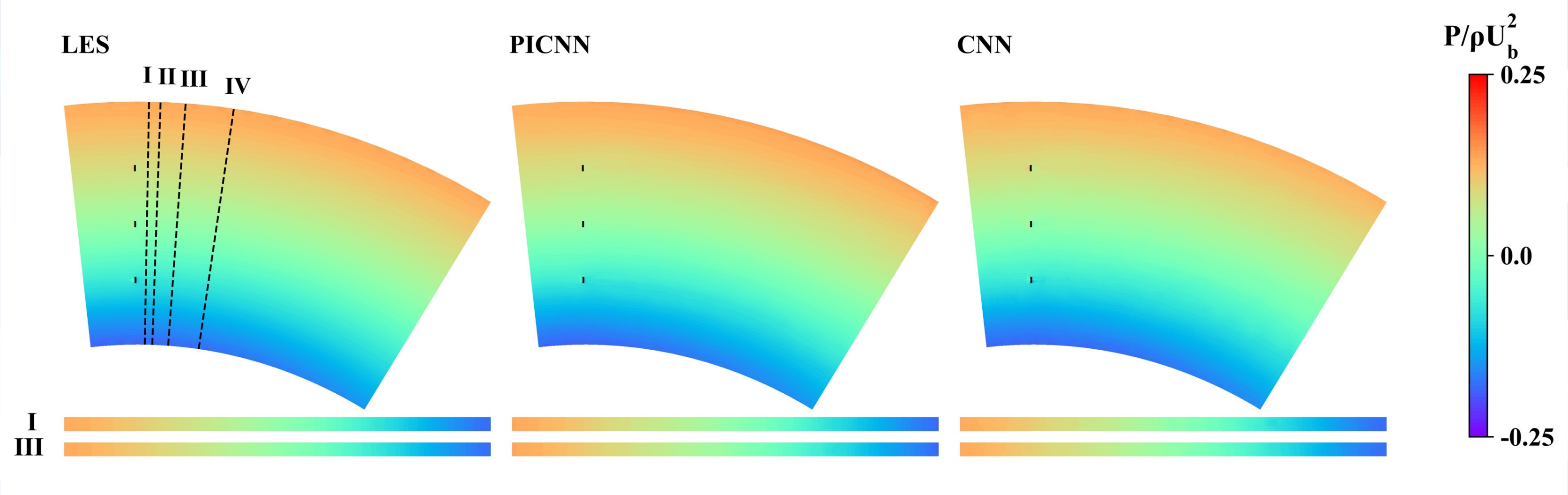
$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$

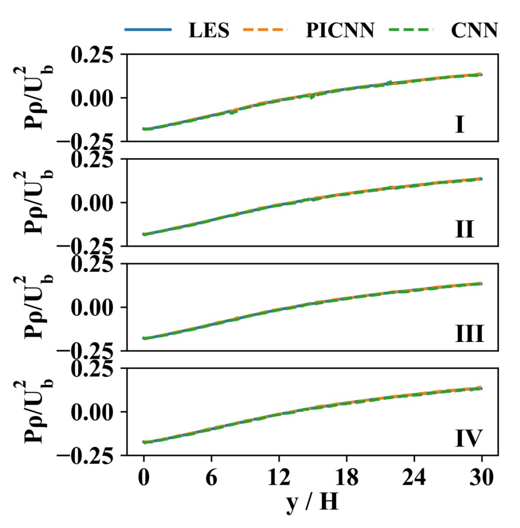






Percentage error:
PICNN: 22.22%
CNN: 38.65%

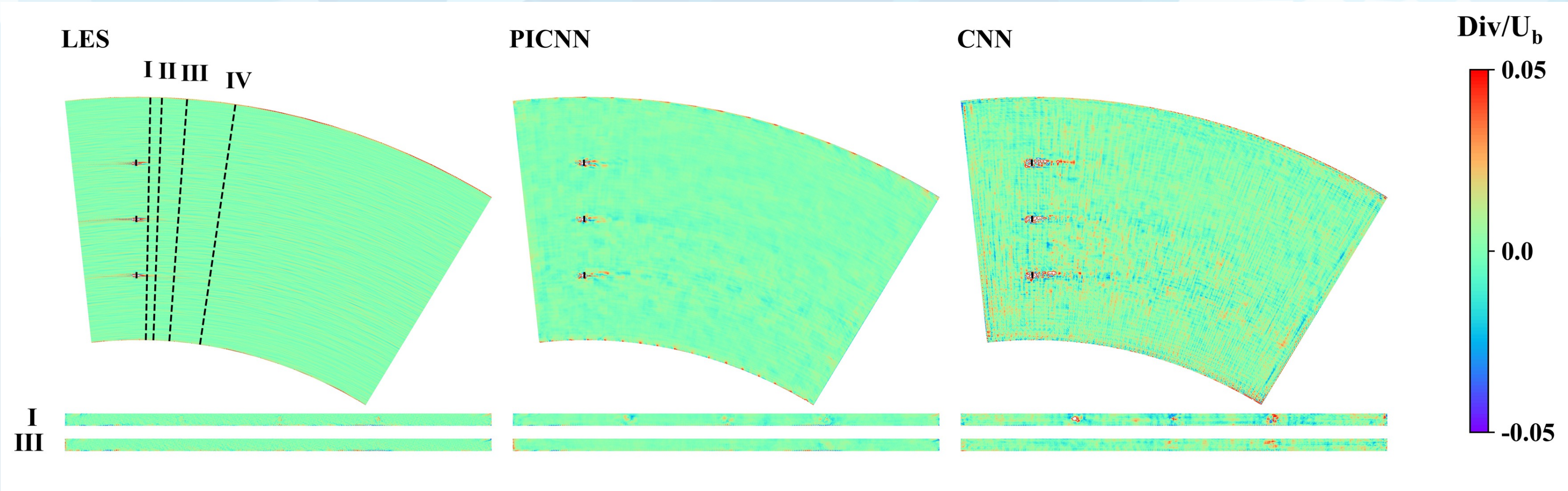


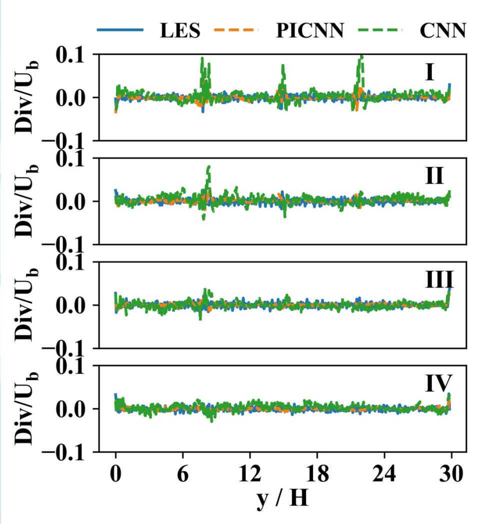


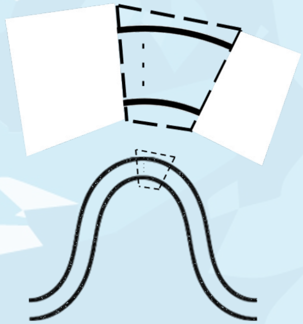
Percentage error:
PICNN: 2.02%
CNN: 5.15%

Results- training case

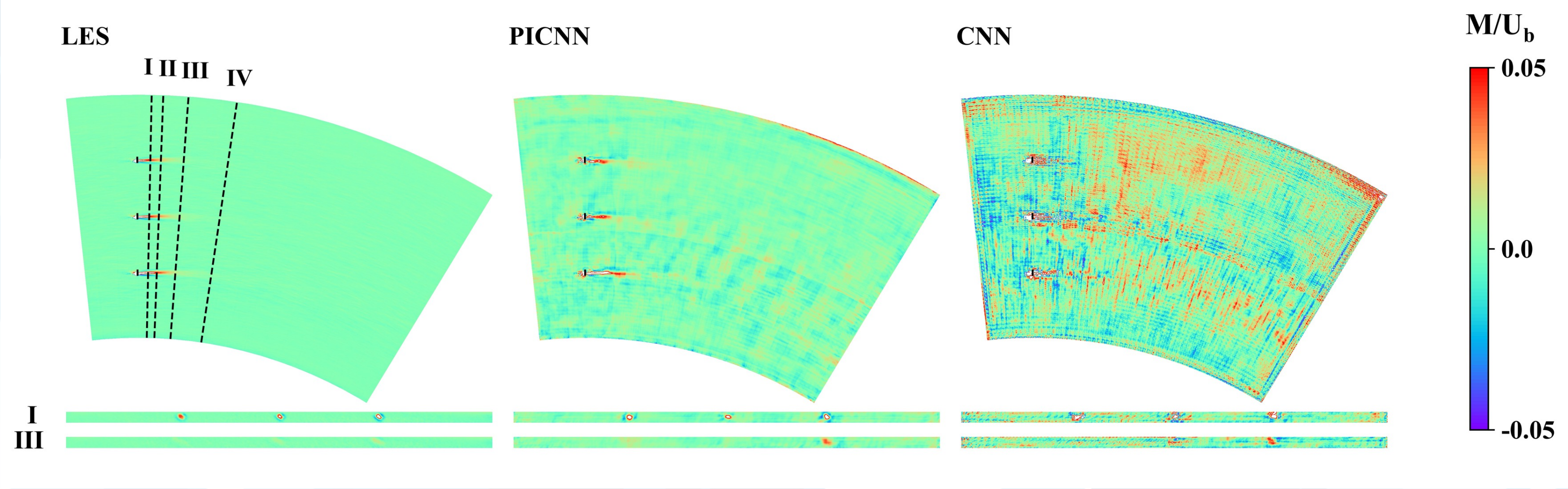
$$\|\Delta\psi\|_2 = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\psi_{i(LES)} - \psi_{i(AI)}\right)^2}$$

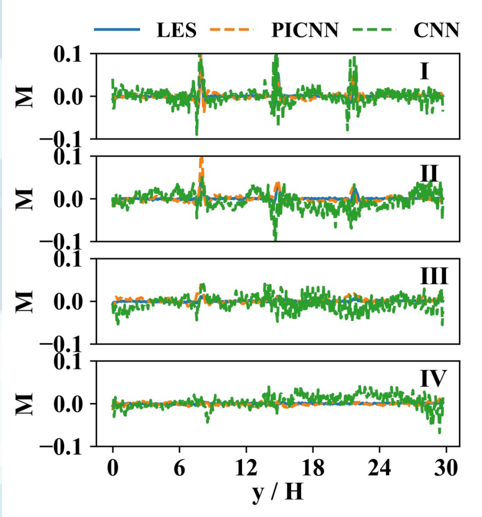






$\|\Delta Div\|_2$:
PICNN: 0.0079
CNN:0.0135

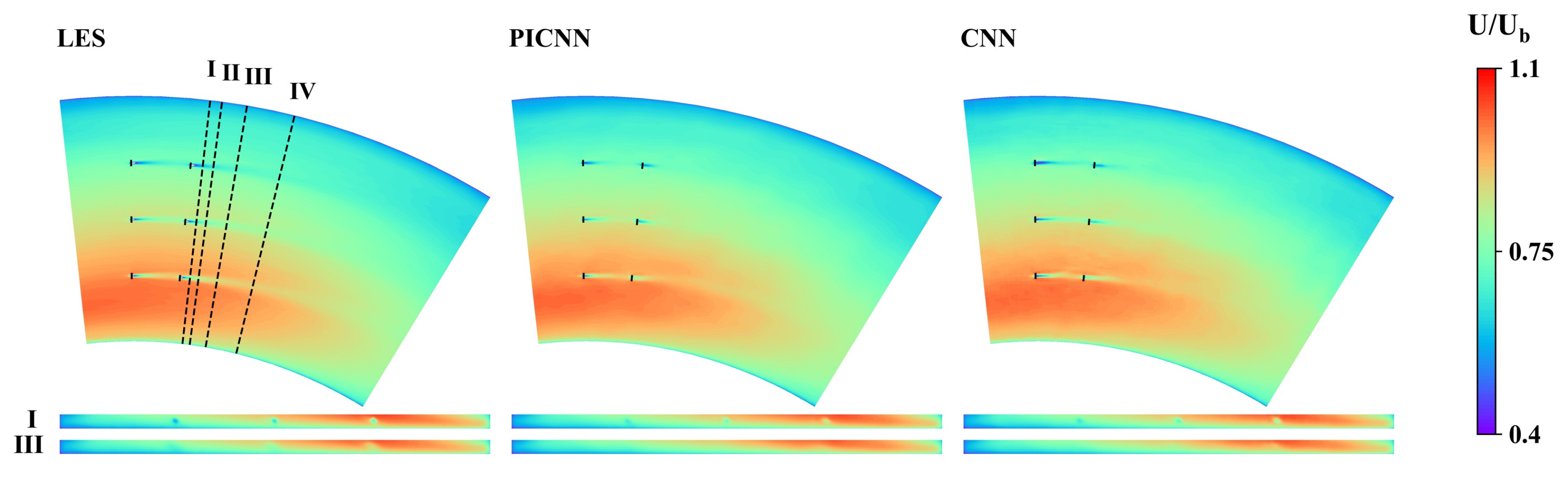


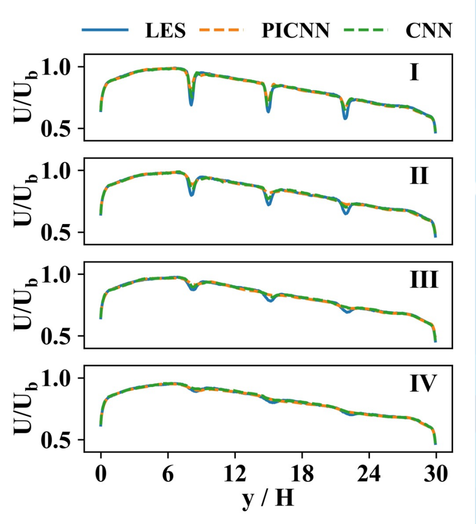


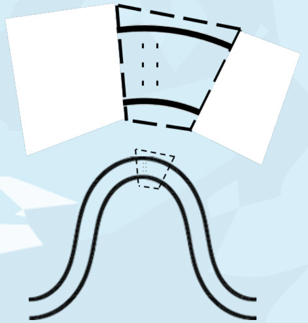
$\|\Delta M\|_2$:
PICNN: 0.0066
CNN: 0.0184

Results- validation case I

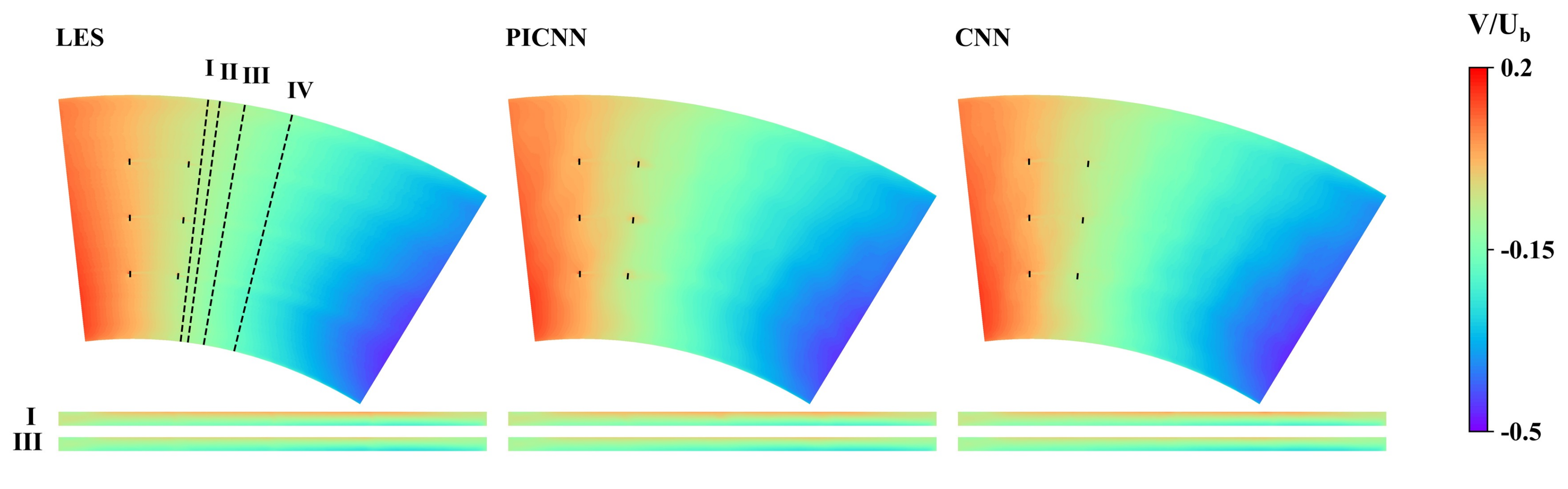
$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$

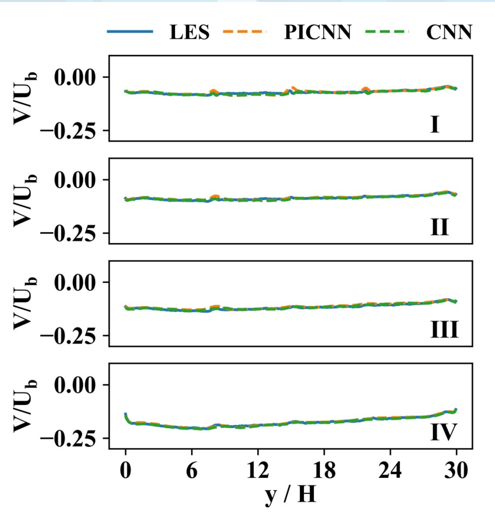






Percentage error:
PICNN: 0.71%
CNN: 0.72%

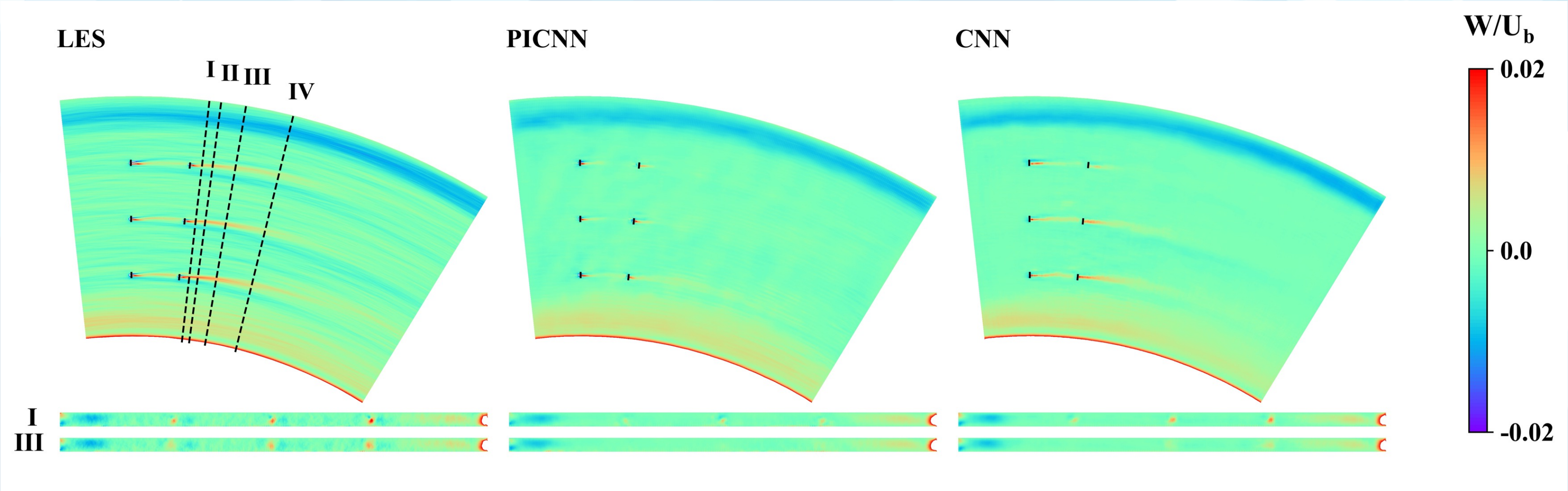


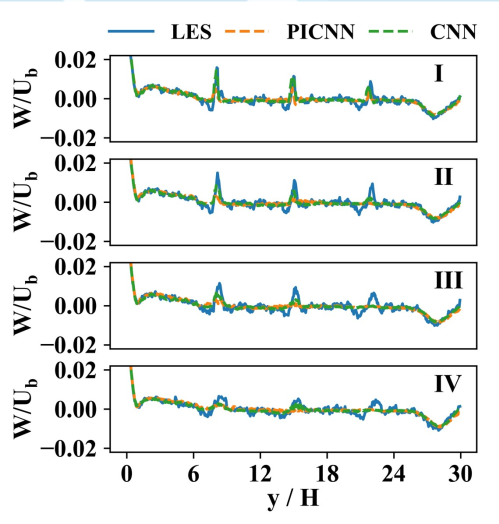


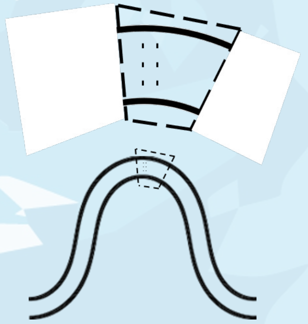
Percentage error:
PICNN: 1.92%
CNN: 2.06%

Results- validation case I

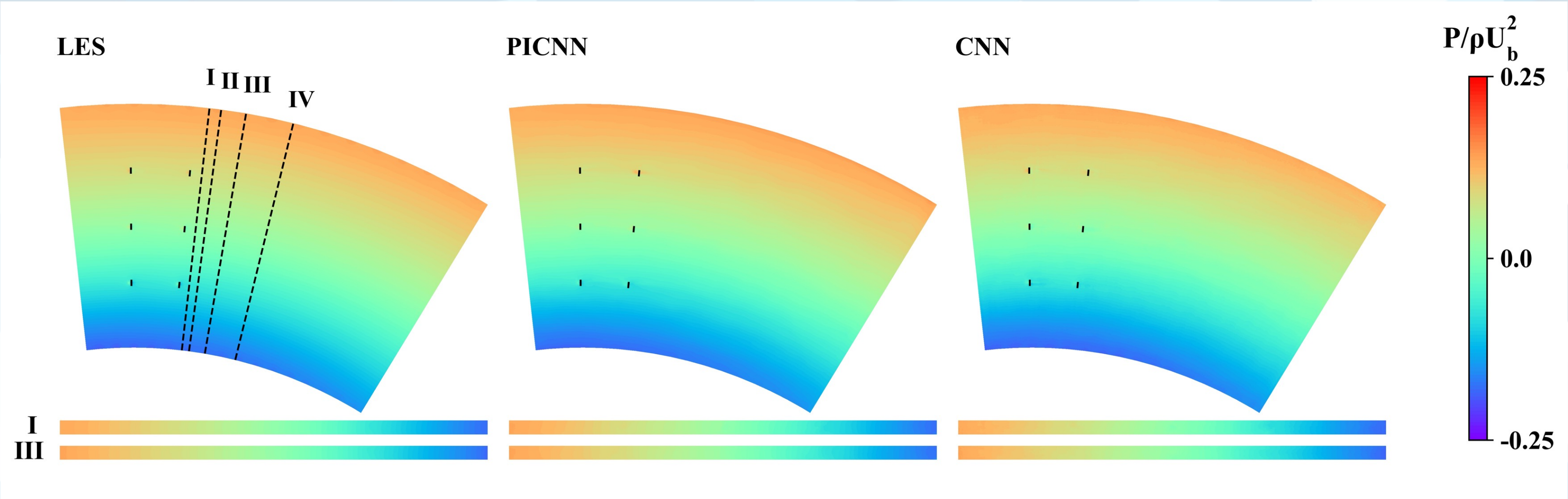
$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$

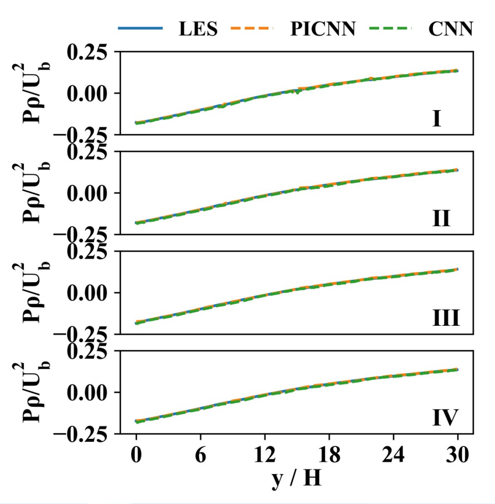






Percentage error:
PICNN: 30.80%
CNN: 44.77%

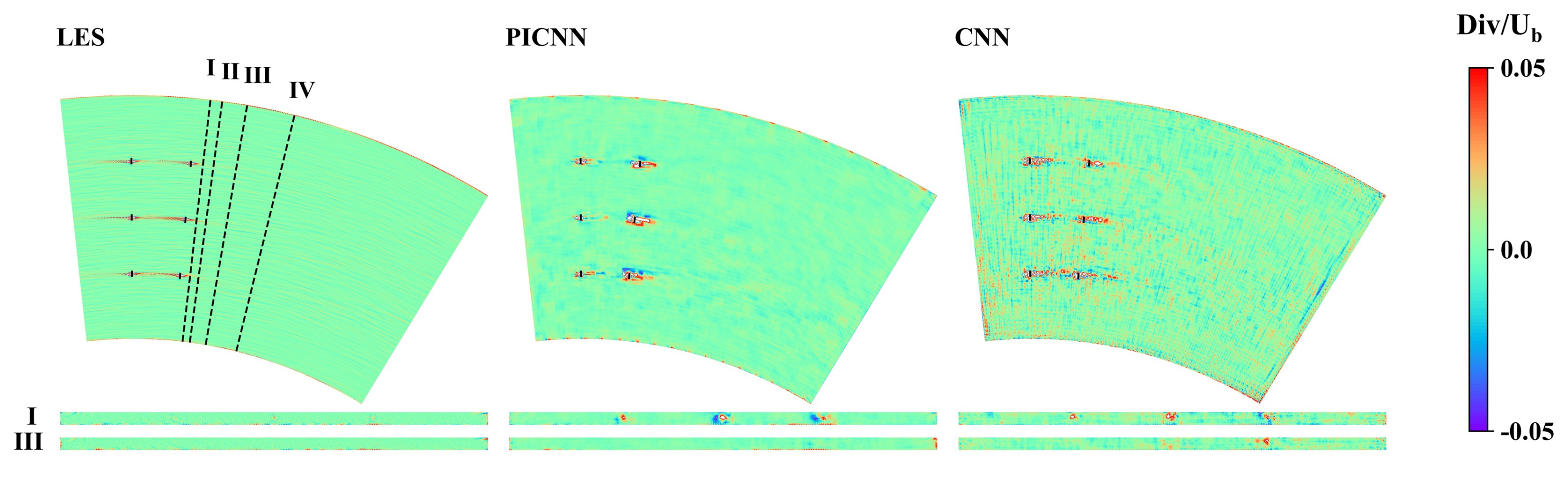


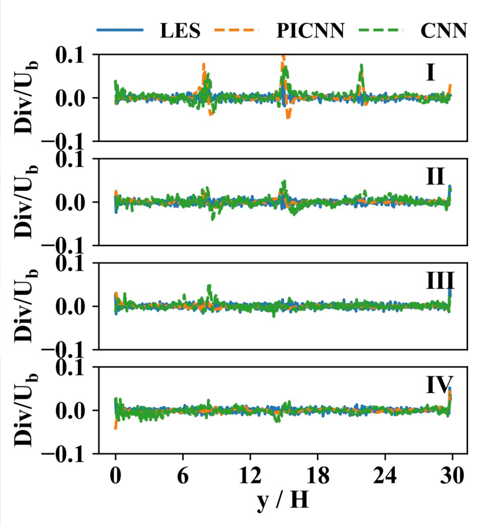


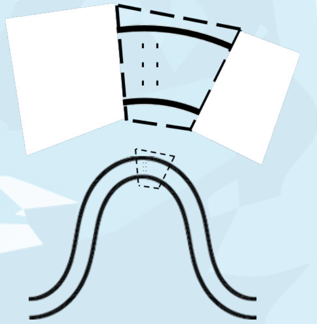
Percentage error:
PICNN: 2.12%
CNN: 5.38%

Results- validation case I

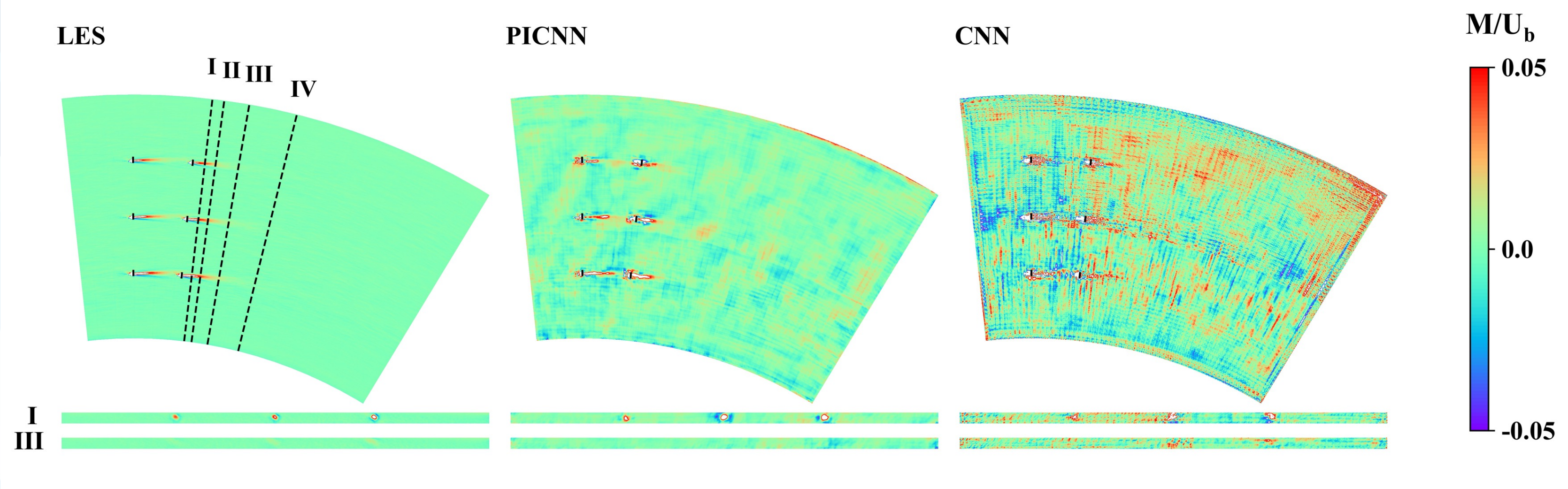
$$\|\Delta\psi\|_2 = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\psi_{i(LES)} - \psi_{i(AI)}\right)^2}$$

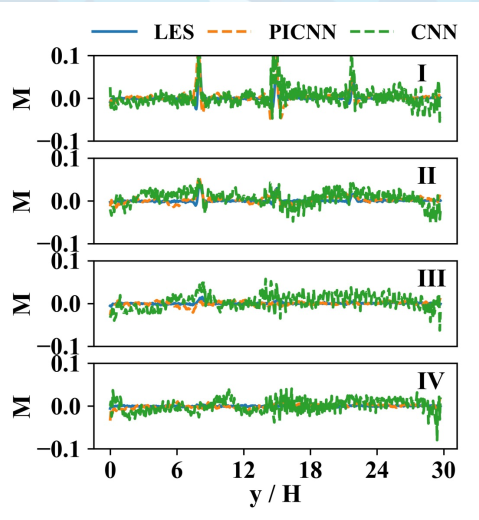






$\|\Delta Div\|_2$:
PICNN: 0.0105
CNN:0.0144

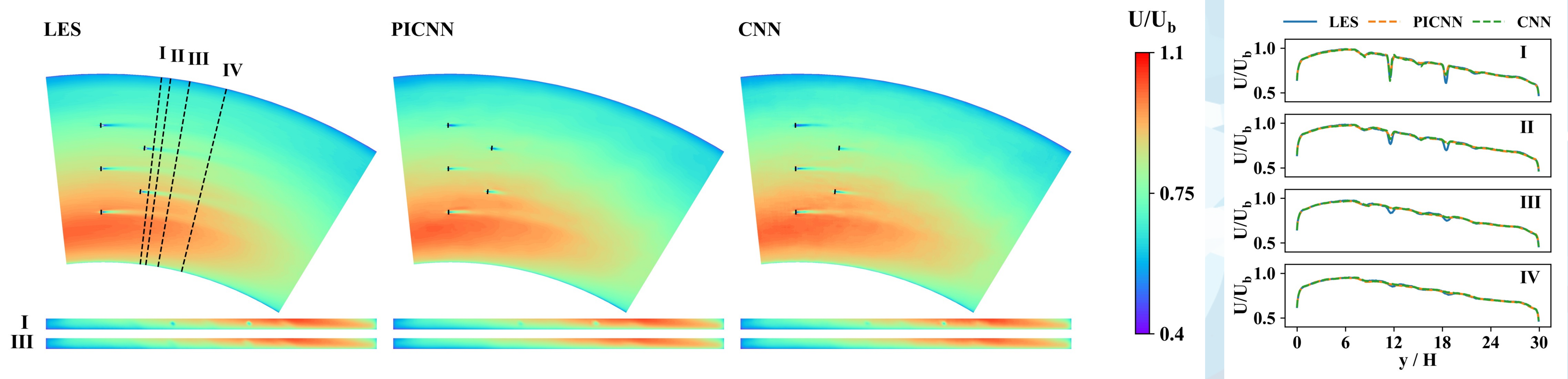


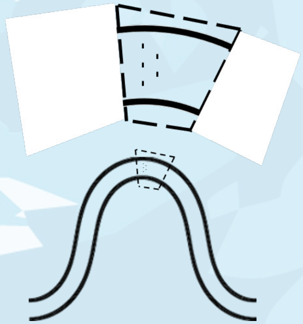


$\|\Delta M\|_2$:
PICNN: 0.0090
CNN: 0.0199

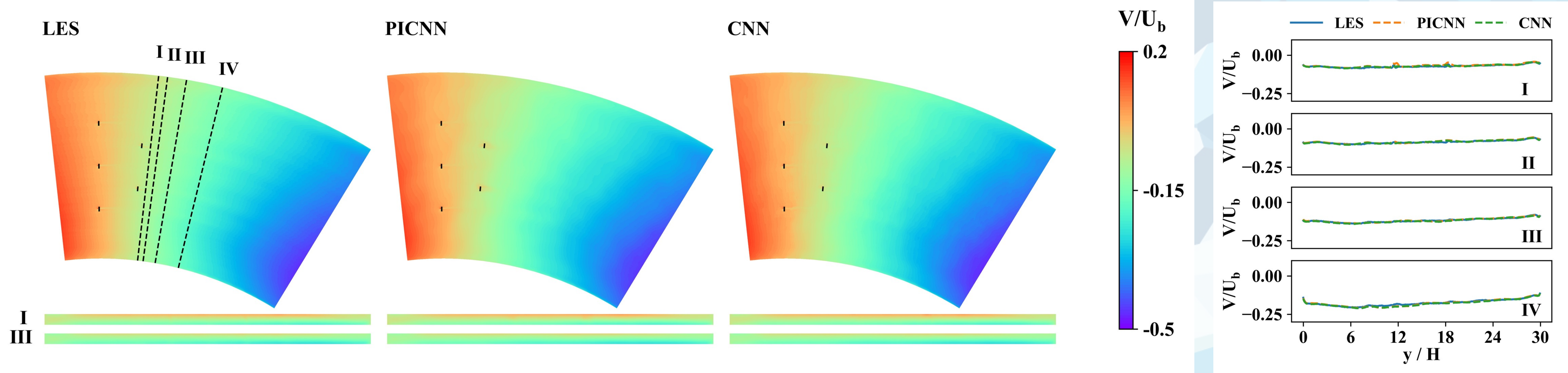
Results- validation case II

$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$





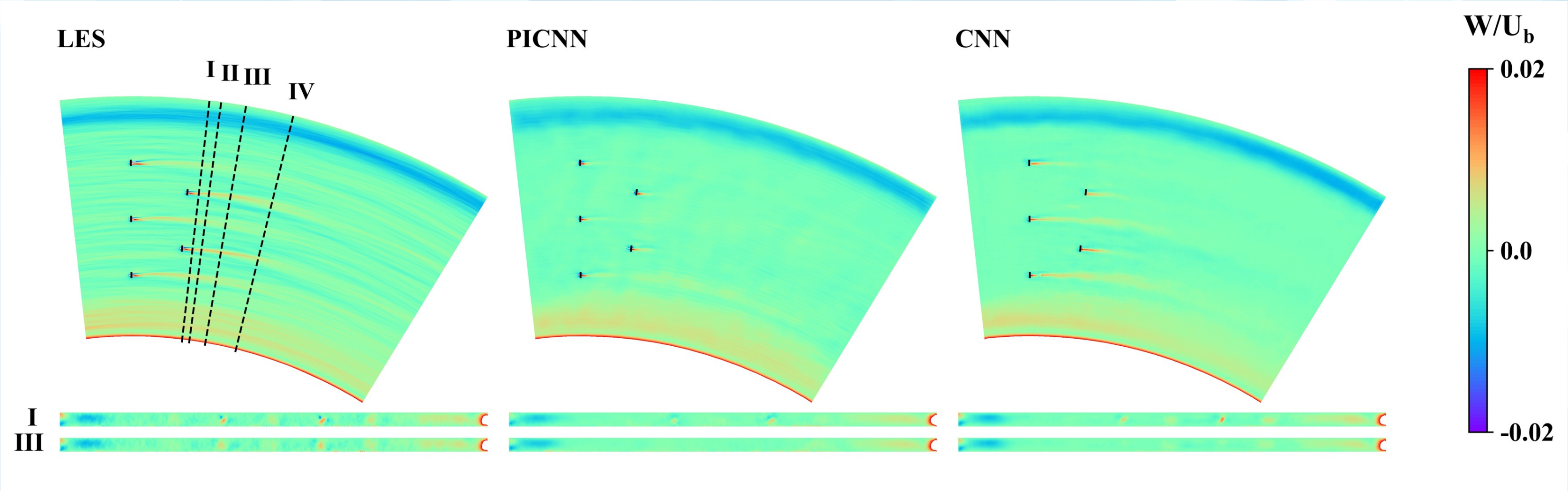
Percentage error:
PICNN: 0.69%
CNN: 0.72%

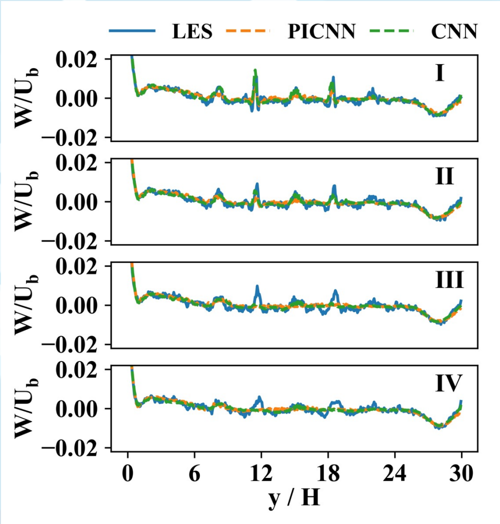


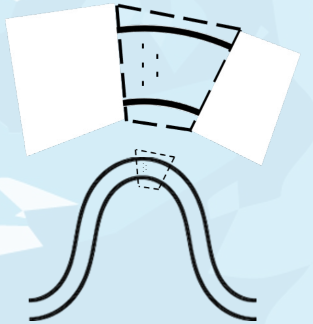
Percentage error:
PICNN: 1.94%
CNN: 2.05%

Results- validation case II

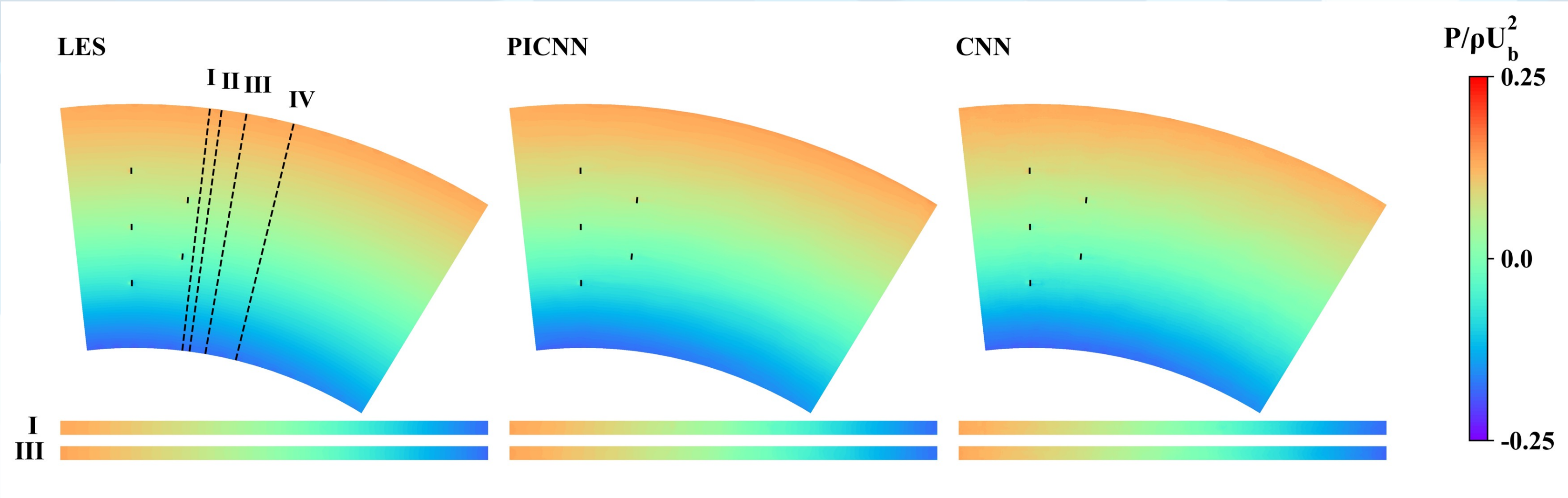
$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$

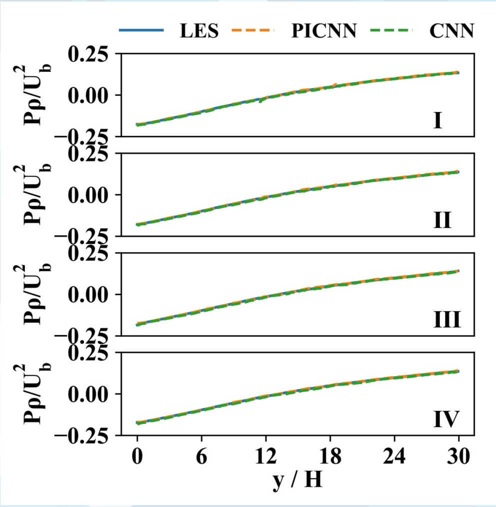






Percentage error:
PICNN: 29.05%
CNN: 43.25%

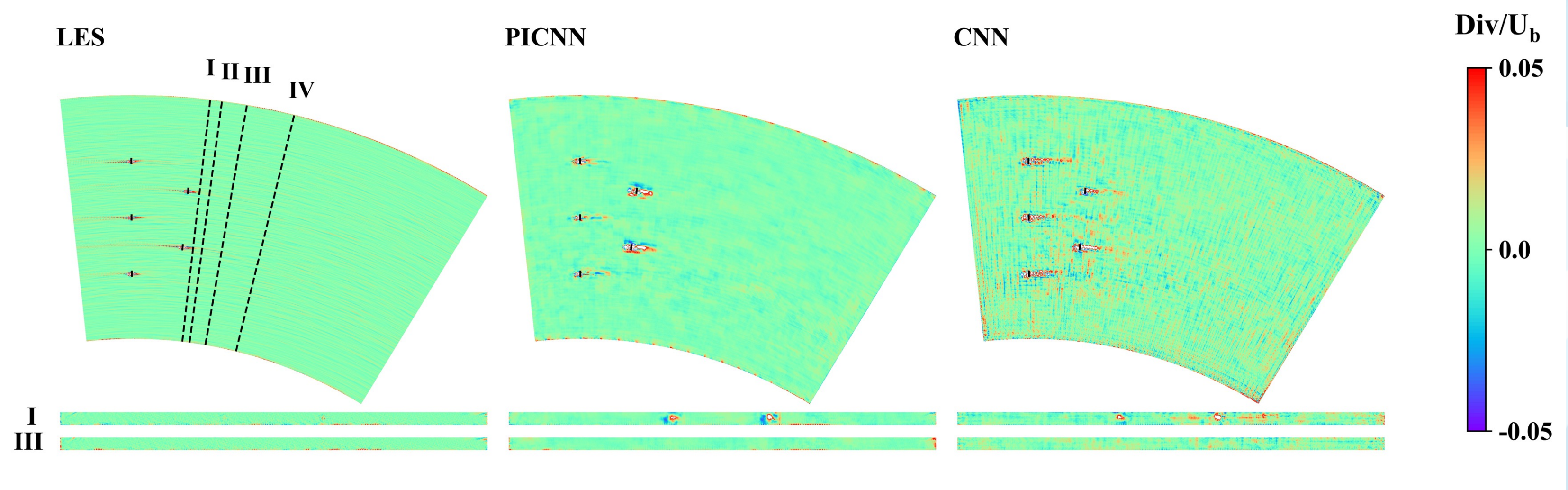


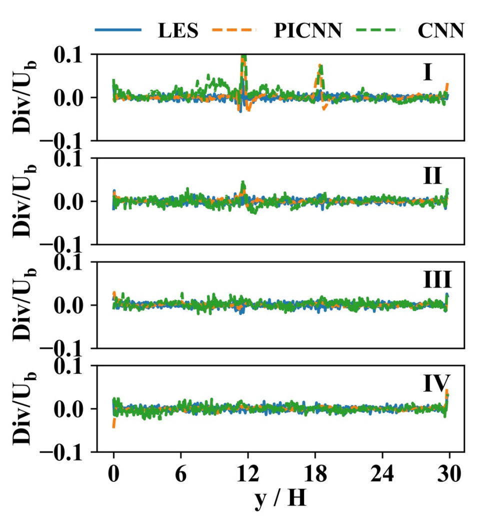


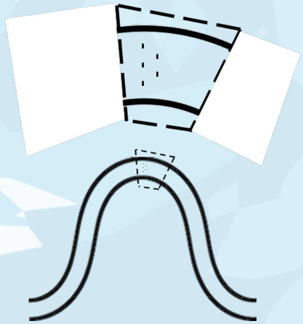
Percentage error:
PICNN: 2.14%
CNN: 5.23%

Results- validation case II

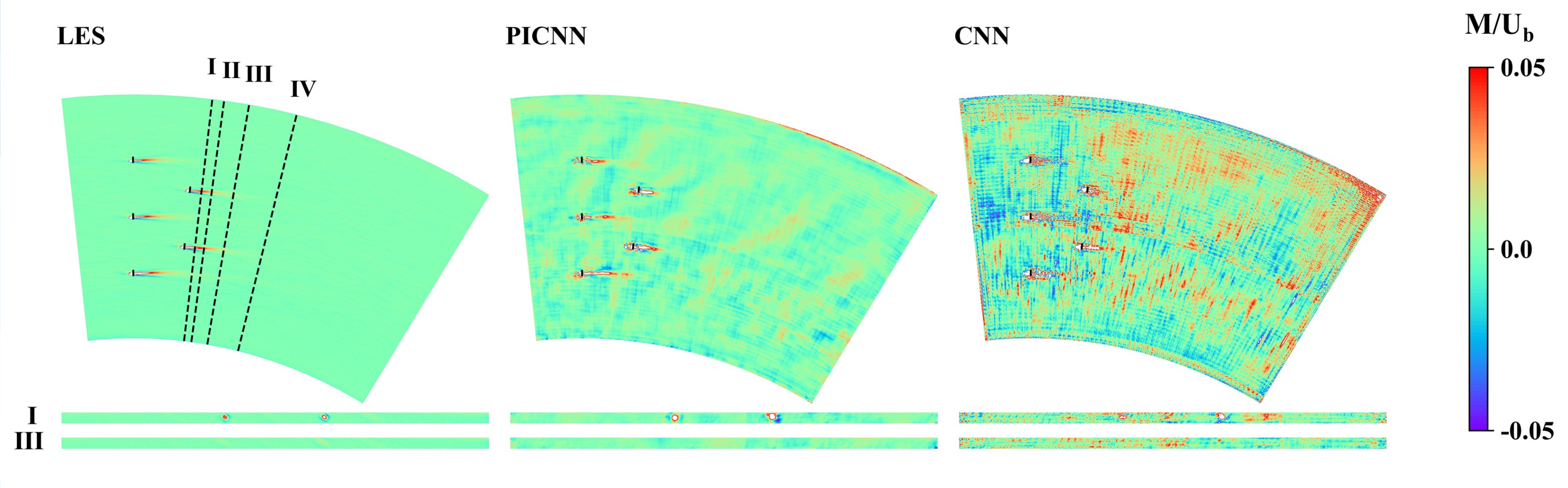
$$\|\Delta\psi\|_2 = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\psi_{i(LES)} - \psi_{i(AI)}\right)^2}$$

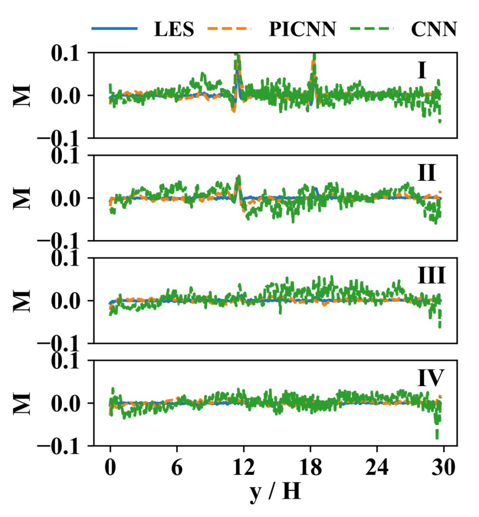






$\|\Delta Div\|_2$:
PICNN: 0.0104
CNN:0.0142





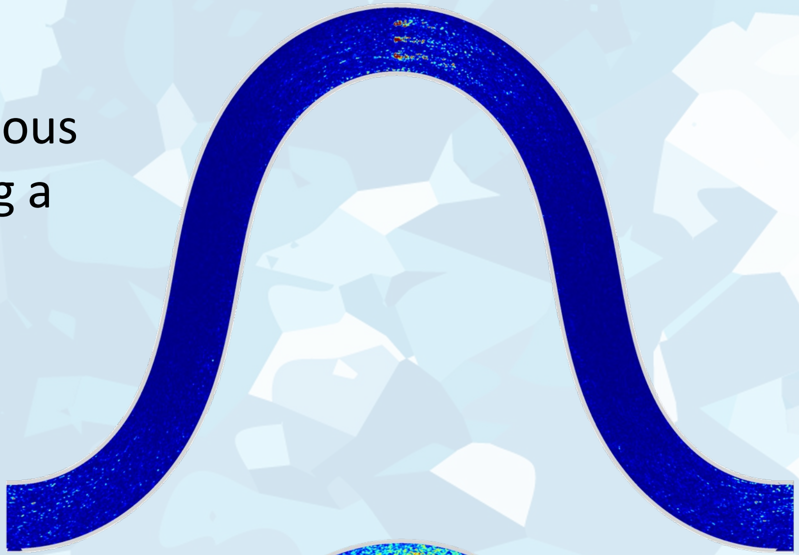
$\|\Delta M\|_2$:
PICNN: 0.0086
CNN: 0.0196

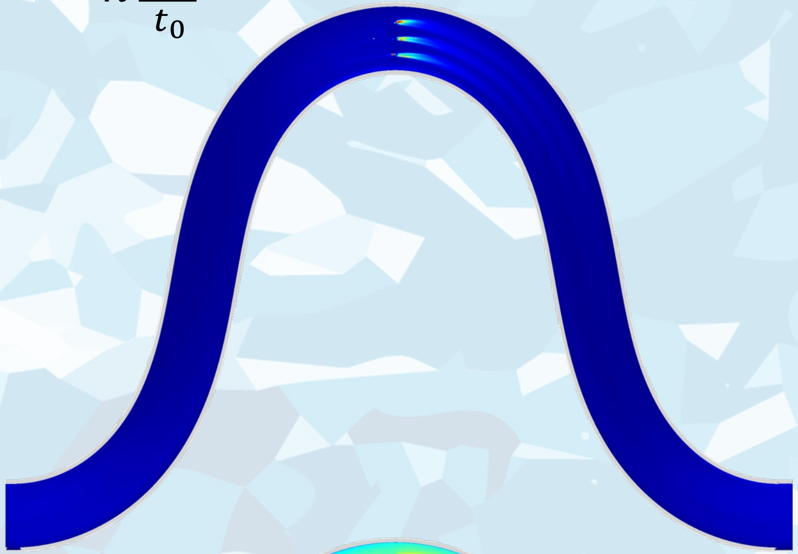
Training- Reynolds stress

$$u'u'_{input} = (u_{LES} - \bar{u}_{CNN})(u_{LES} - \bar{u}_{CNN})$$

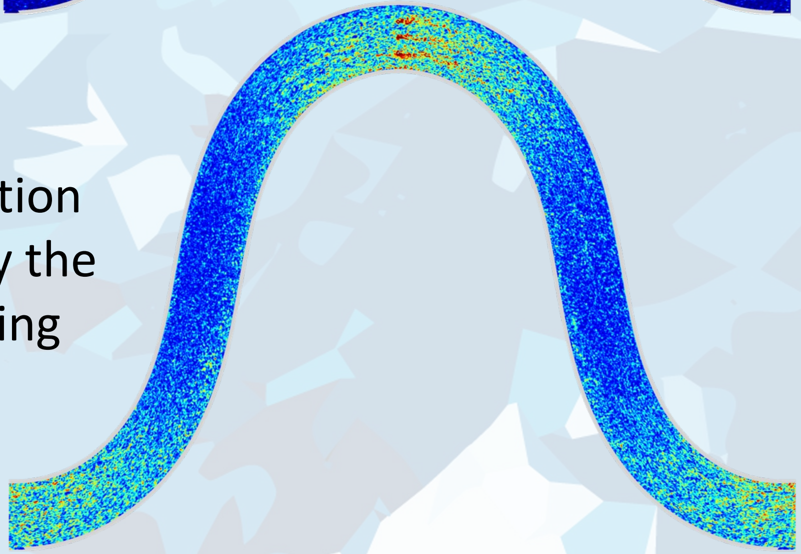
$$\overline{u'u'}_{output} = \frac{1}{n}\sum_{t_0}^{t_n}(u_{LES} - \bar{u}_{LES})(u_{LES} - \bar{u}_{LES})$$

Input is the instantaneous Reynolds stress, having a highly heterogeneous distribution

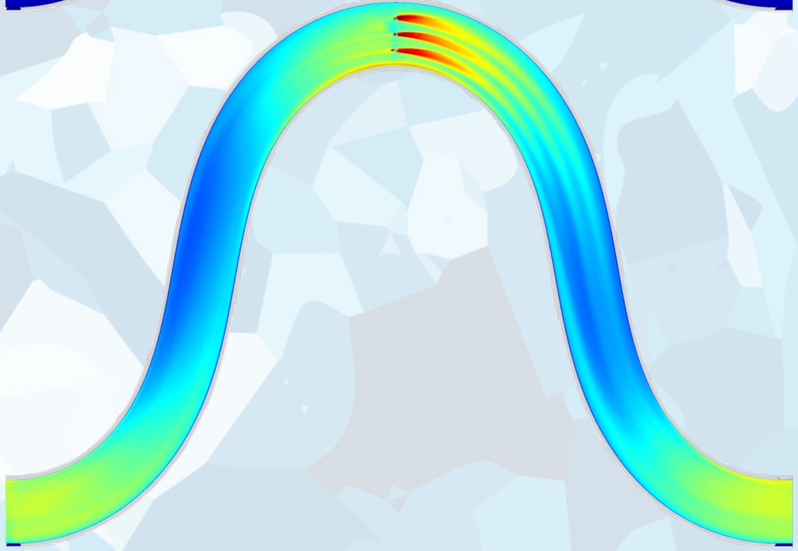




Rendering the distribution more homogeneous by the cubic root pre-processing

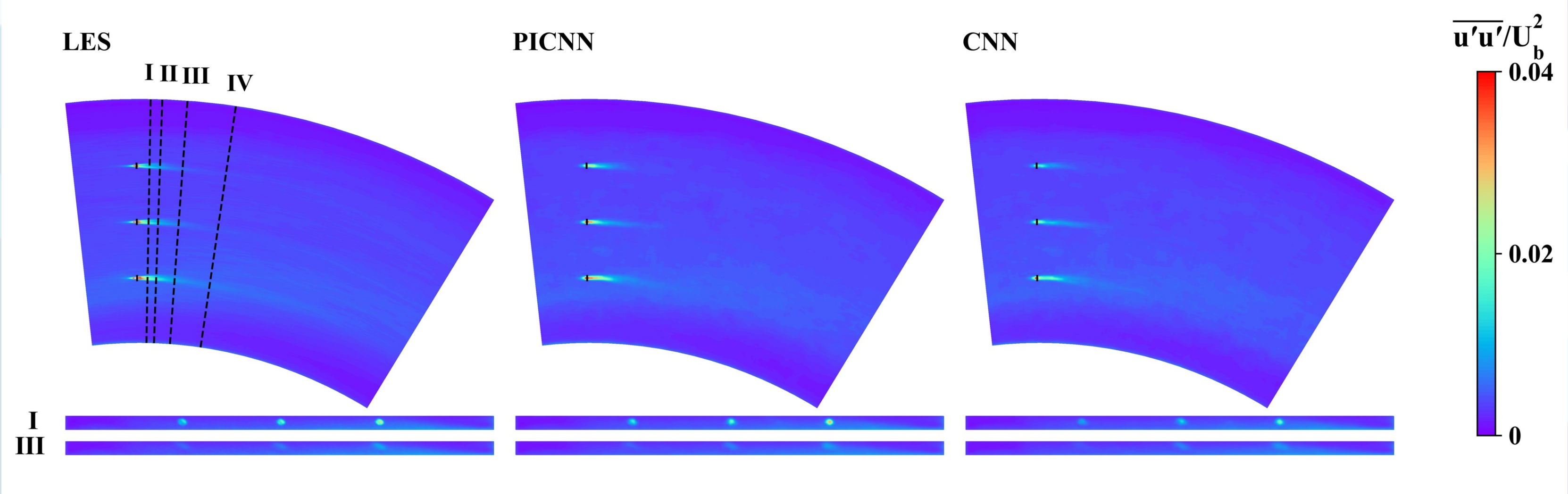


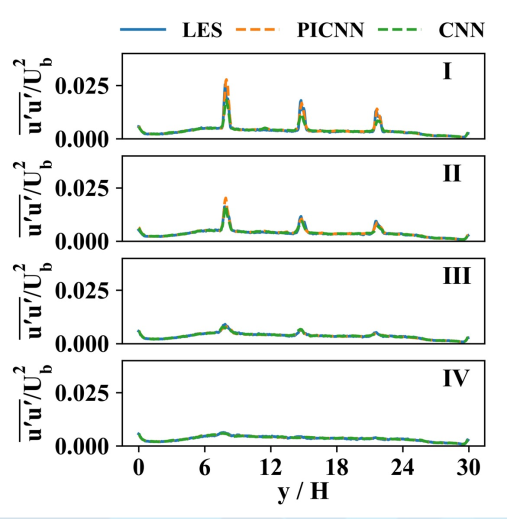
CNN

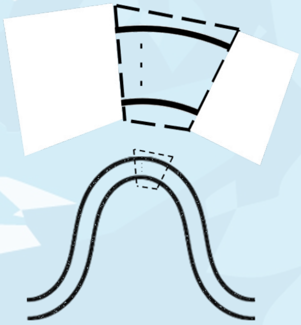


Results- training case

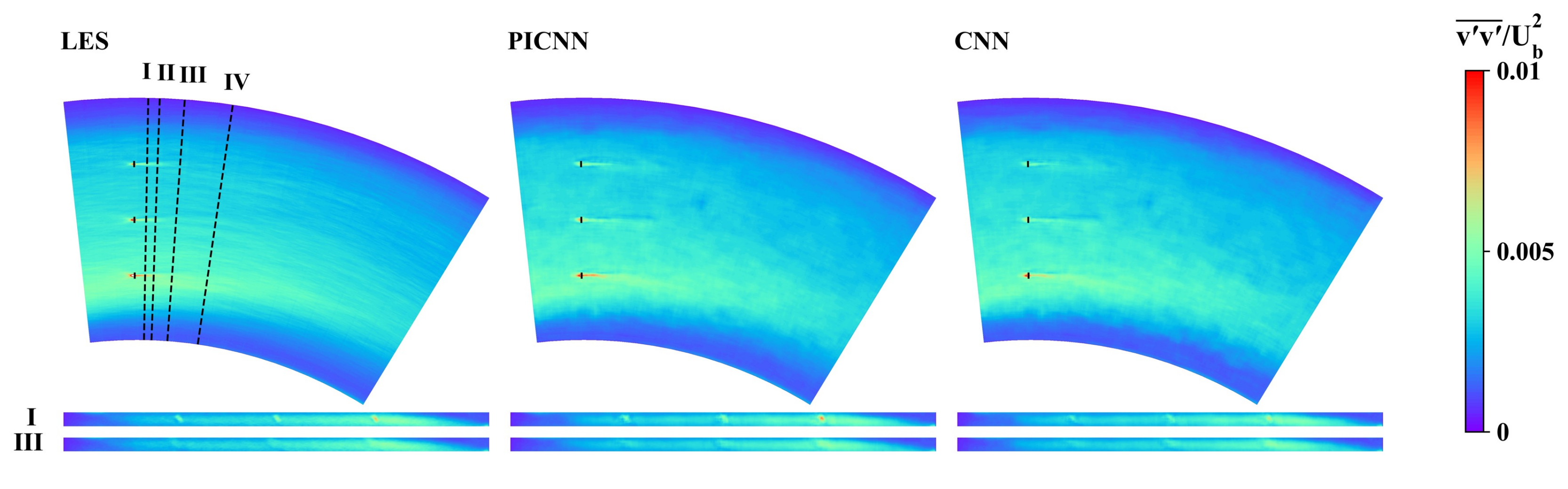
$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$

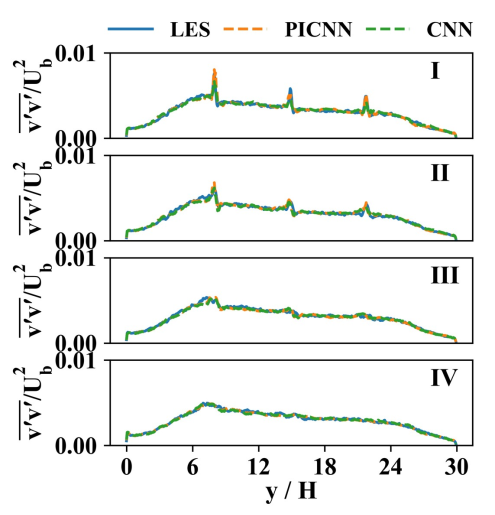






Percentage error:
PICNN: 5.33%
CNN: 6.04%

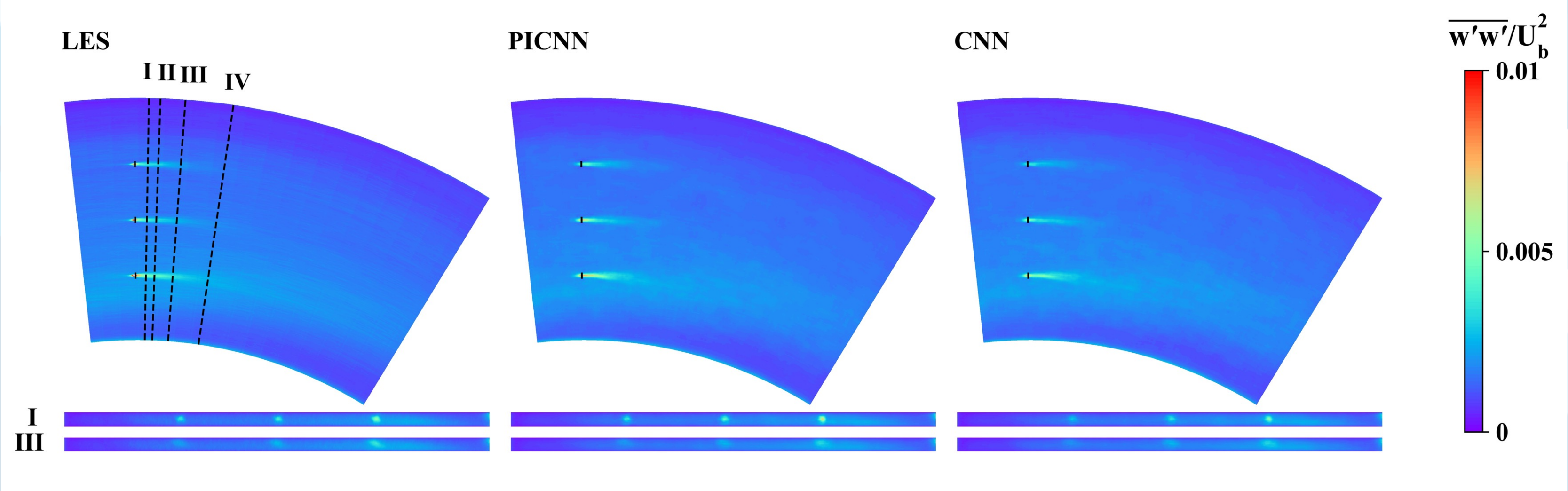


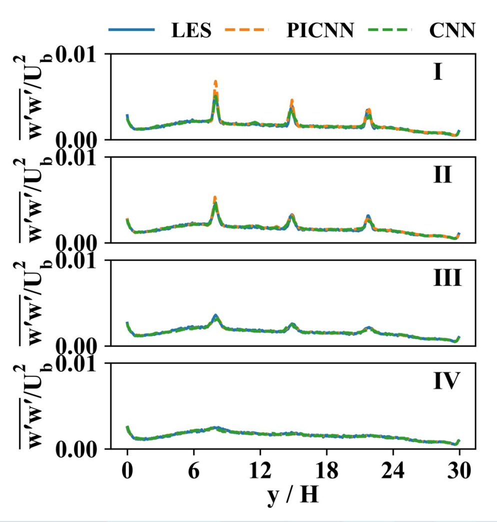


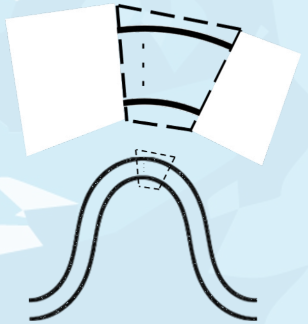
Percentage error:
PICNN: 4.52%
CNN: 5.00%

Results- training case

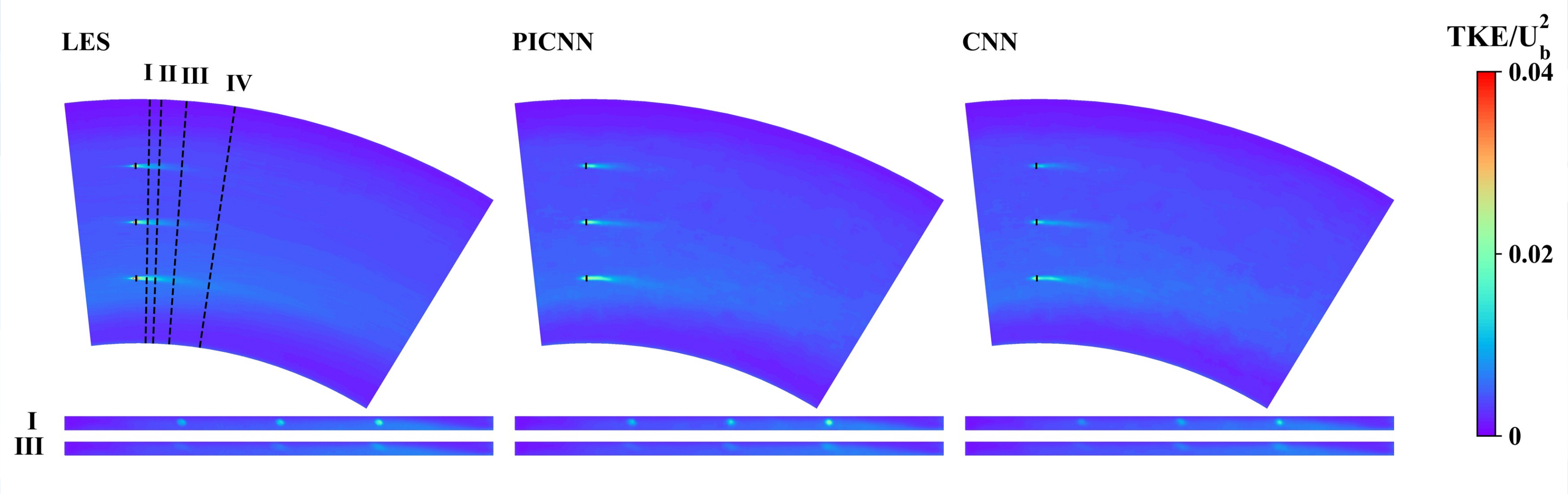
$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$

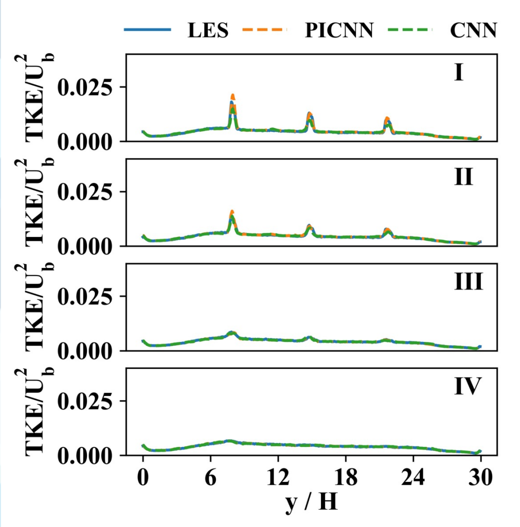






Percentage error:
PICNN: 4.70%
CNN: 5.05%

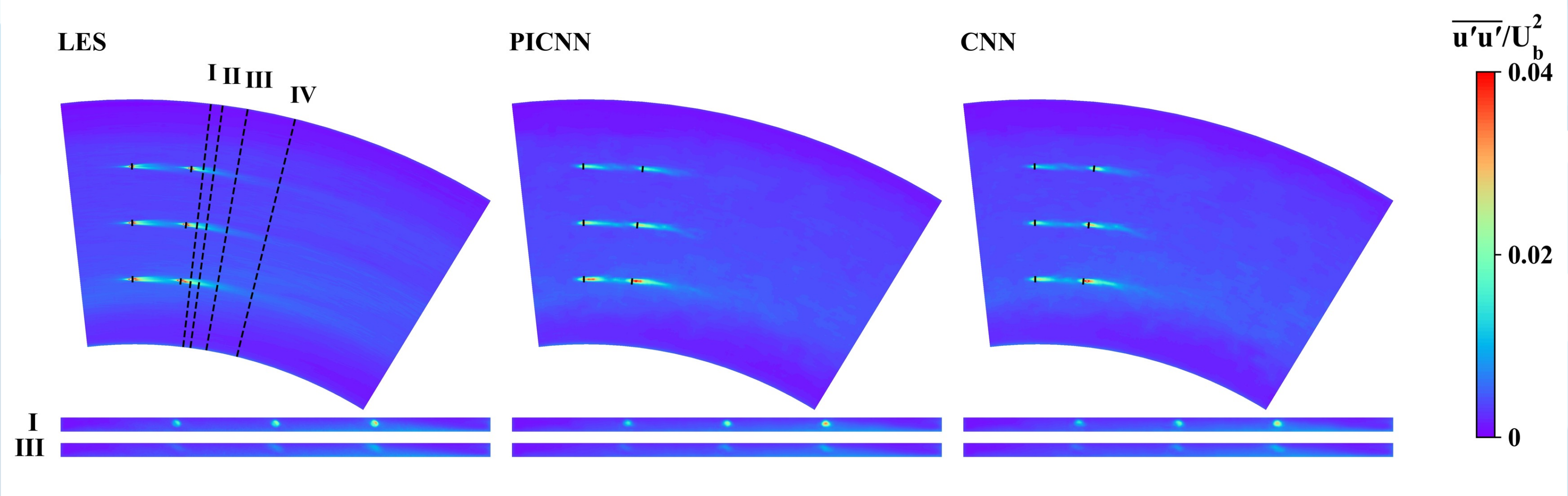


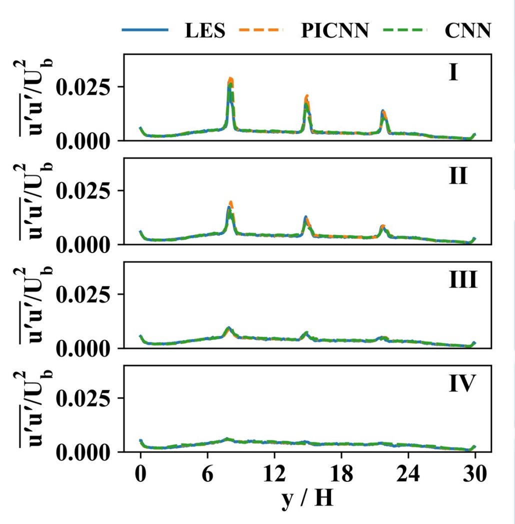


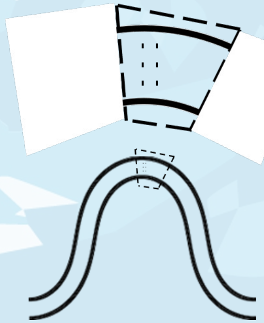
Percentage error:
PICNN: 3.93%
CNN: 4.51%

Results- validation case I

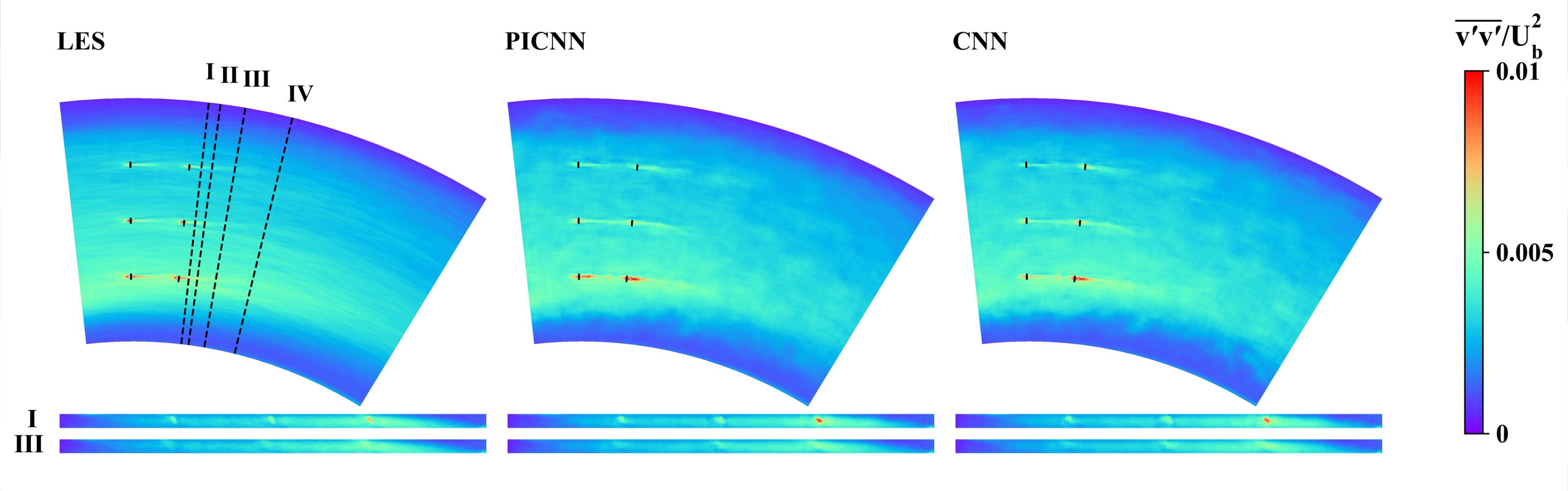
$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$

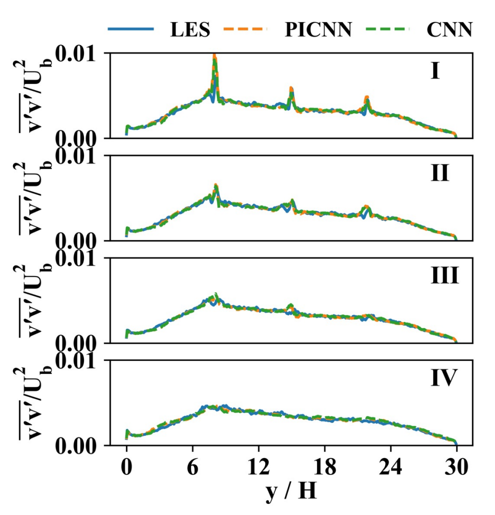






Percentage error:
PICNN: 6.80%
CNN: 7.25%

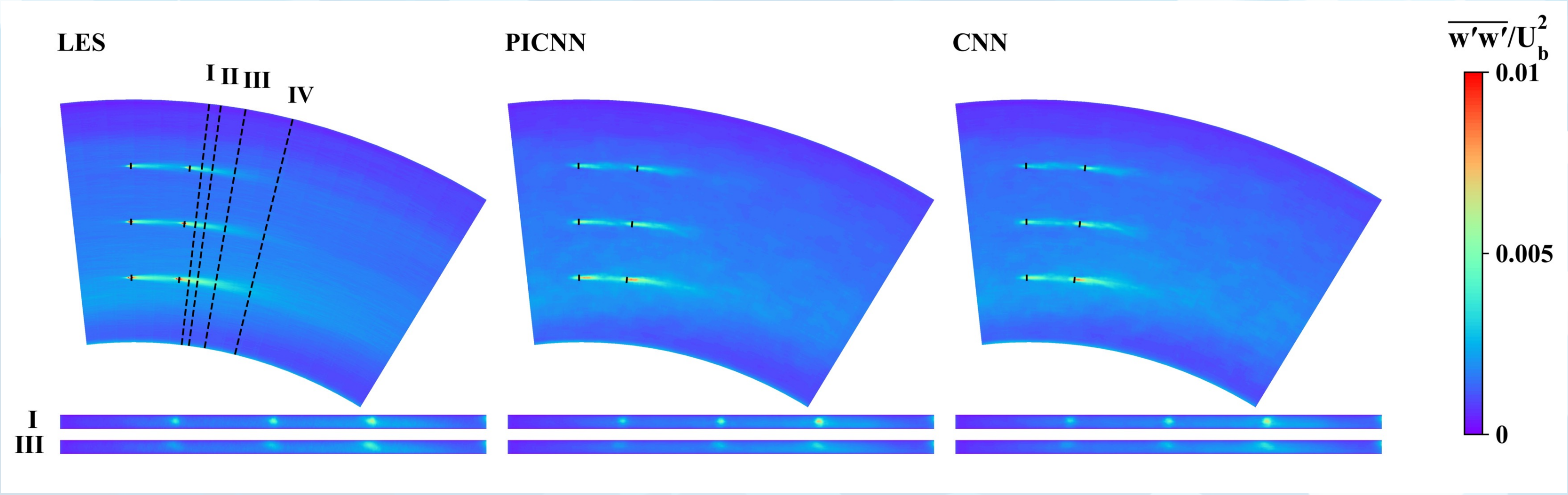


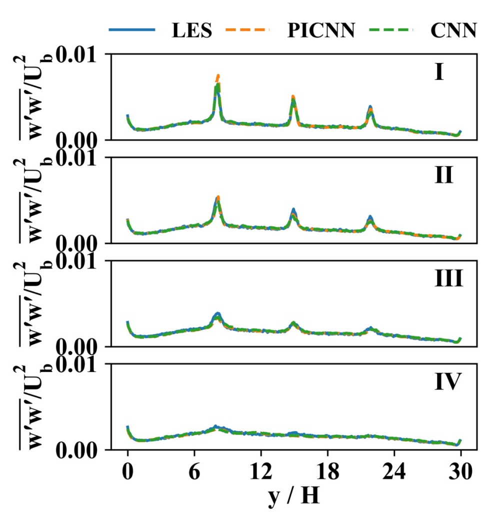


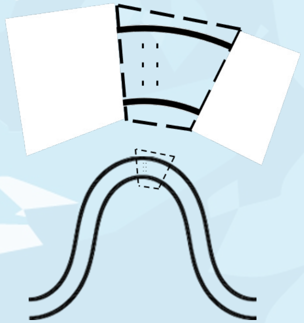
Percentage error:
PICNN: 5.83%
CNN: 5.85%

Results- validation case I

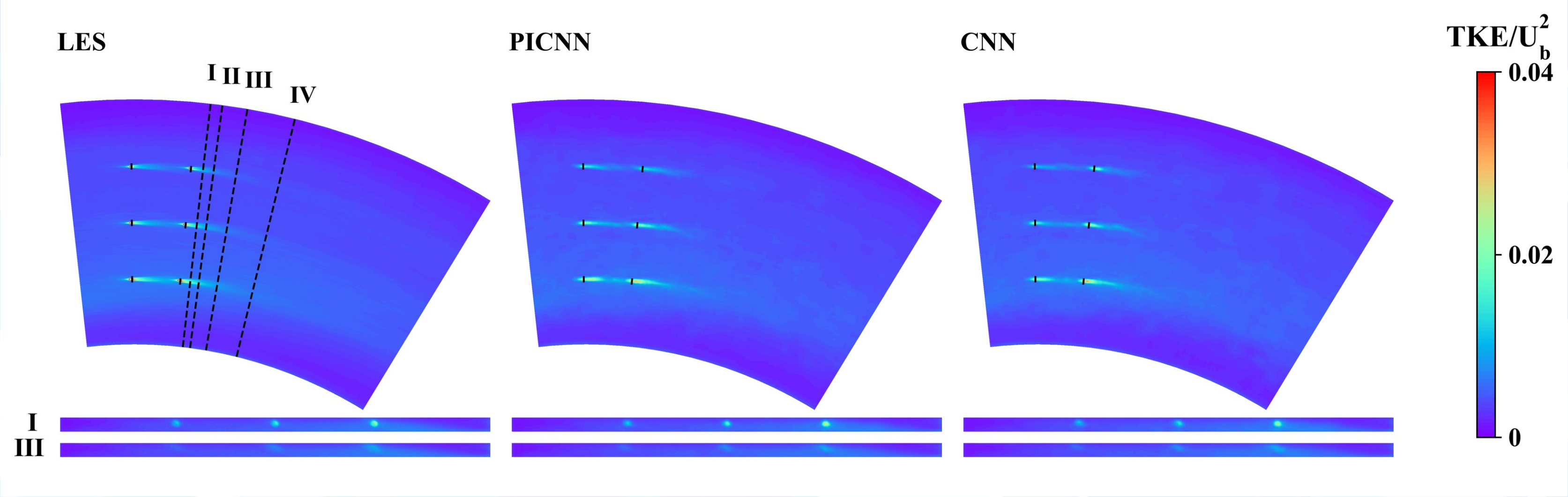
$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$

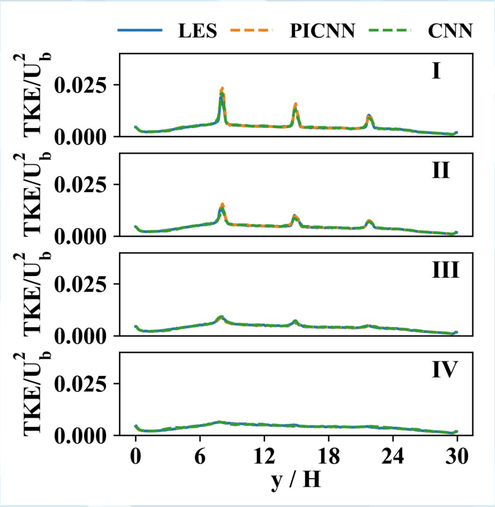






Percentage error:
PICNN: 5.66%
CNN: 5.68%

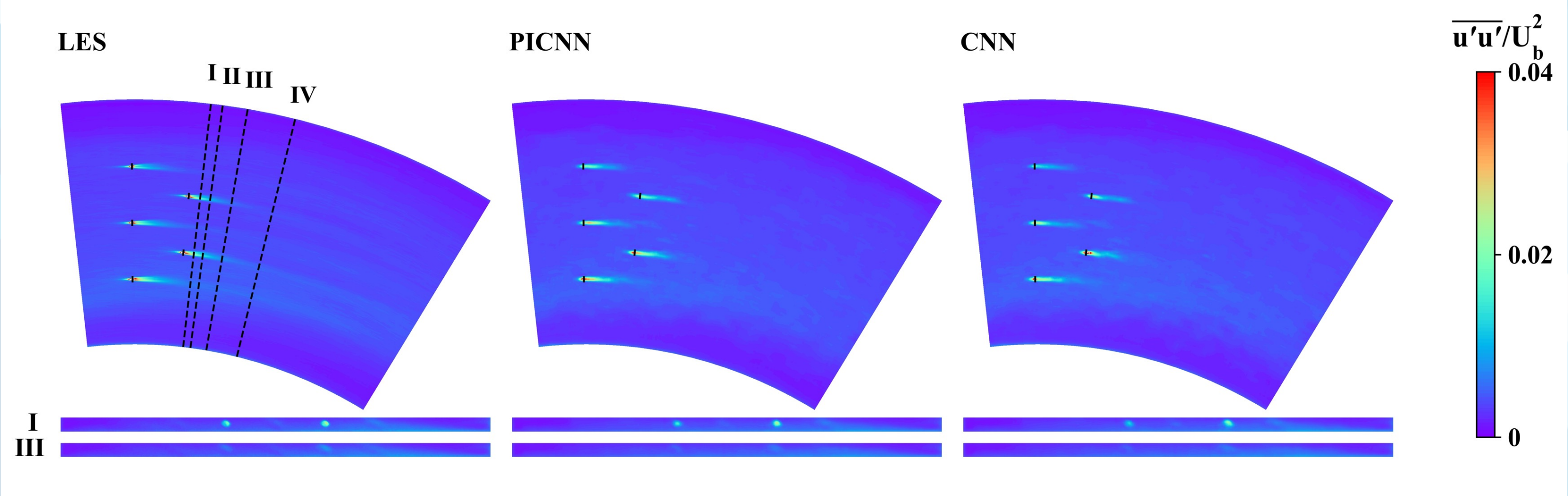


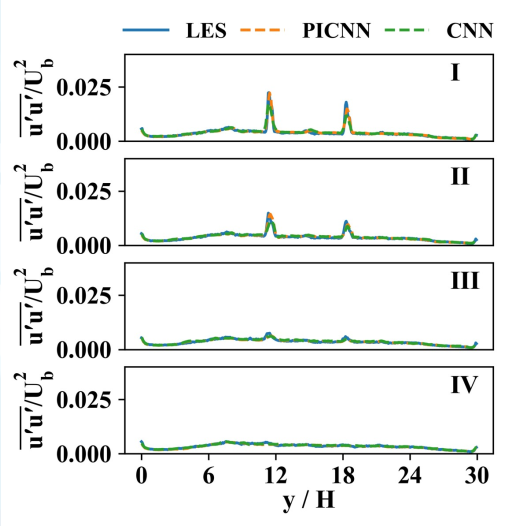


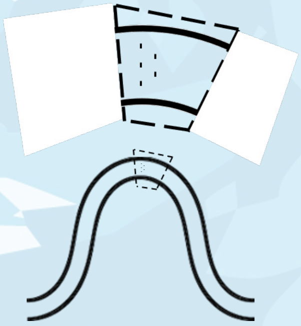
Percentage error:
PICNN: 5.21%
CNN: 5.33%

Results- validation case II

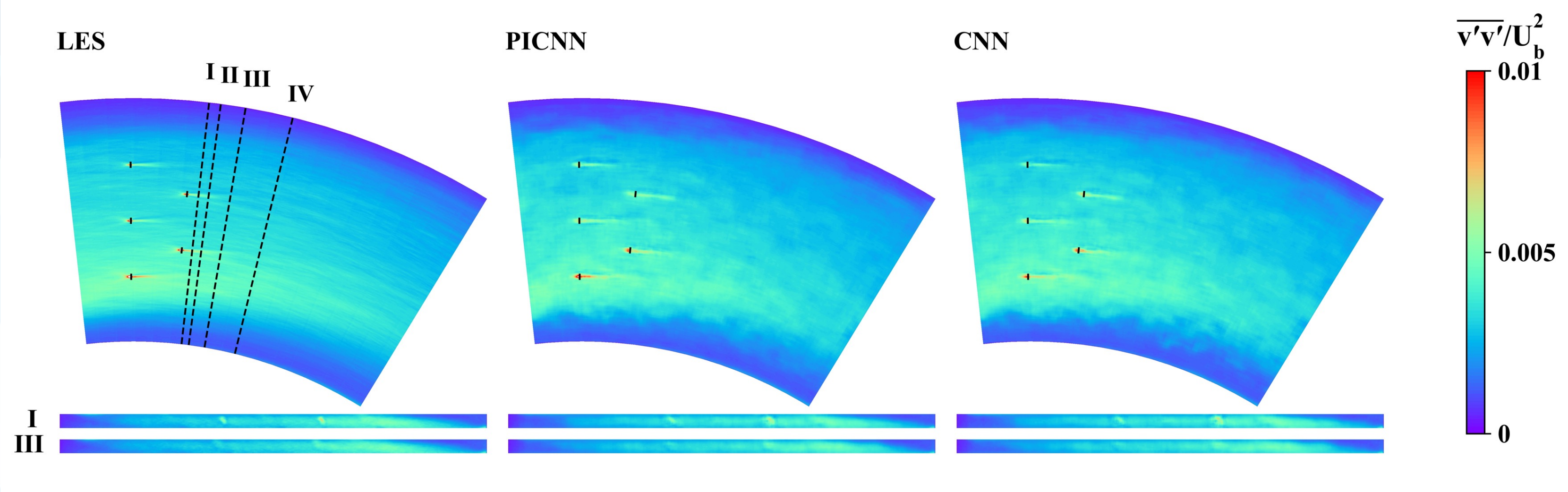
$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$

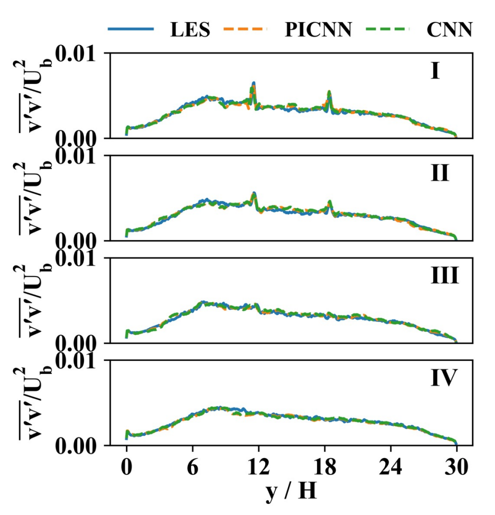






Percentage error:
PICNN: 6.61%
CNN: 7.10%

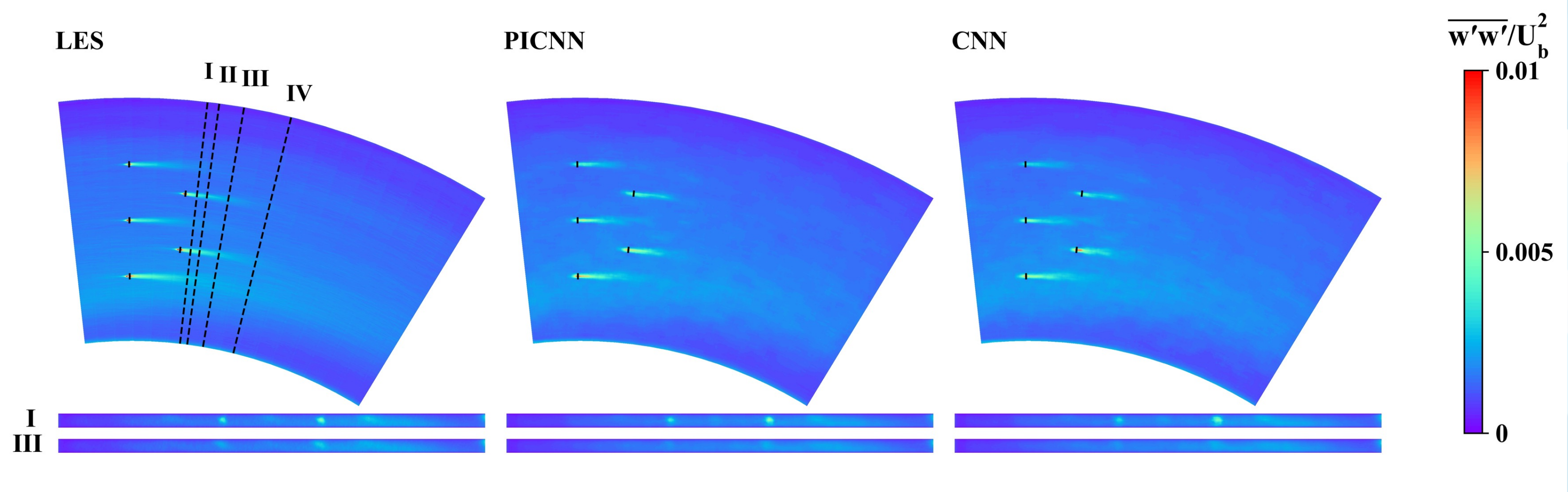


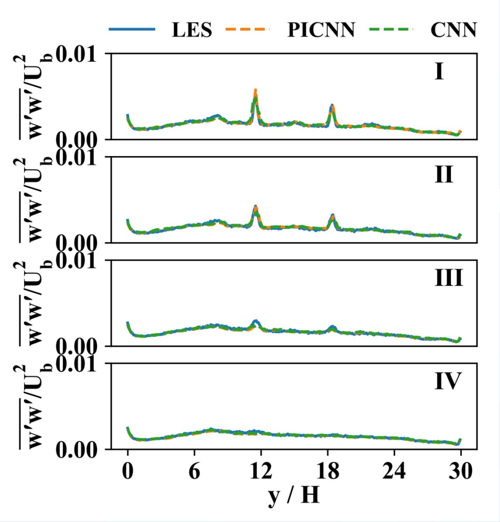


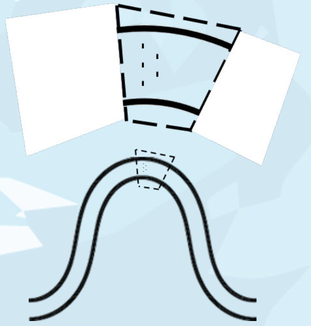
Percentage error:
PICNN: 5.57%
CNN: 5.73%

Results- validation case II

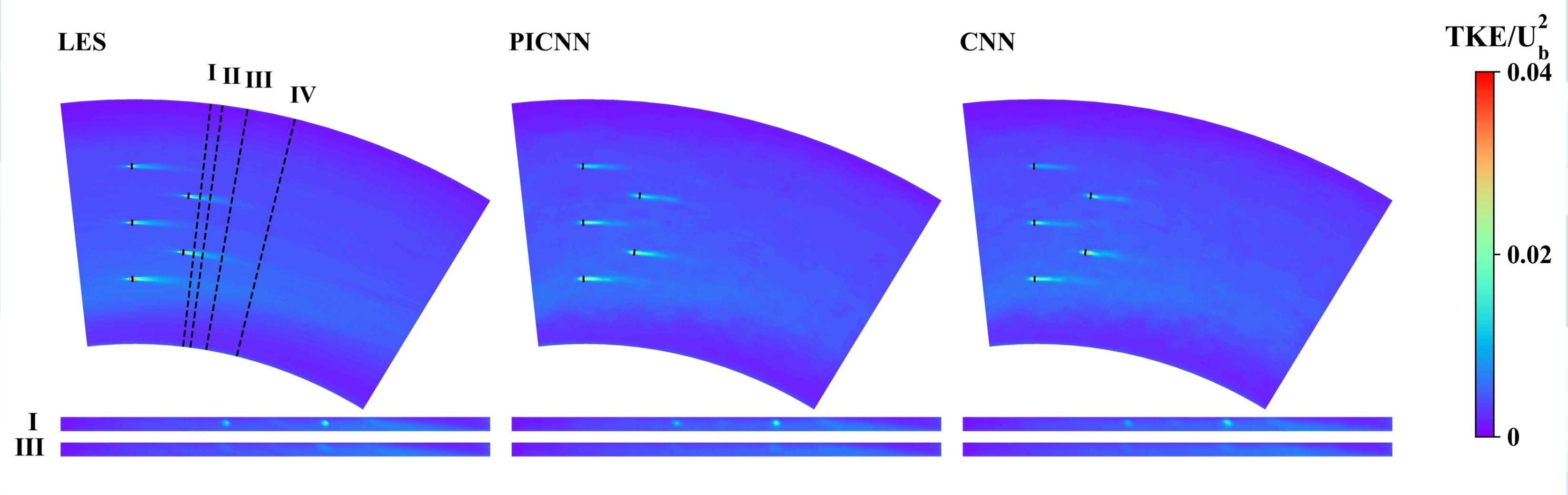
$$\text{Percentage error} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)} - \psi_{i(AI)}\right|}{\frac{1}{N}\sum_{i=1}^{N}\left|\psi_{i(LES)}\right|}$$

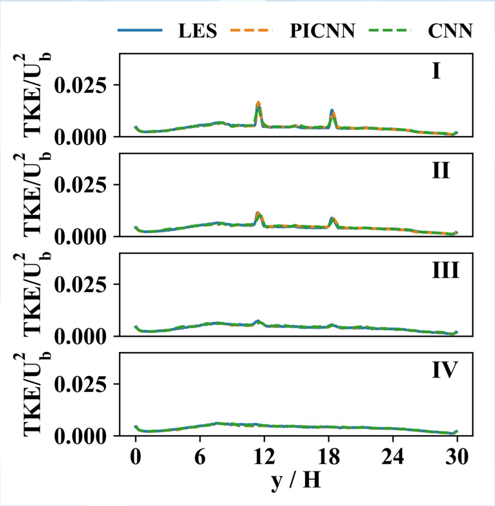






Percentage error:
PICNN: 5.43%
CNN: 5.65%





Percentage error:
PICNN: 5.03%
CNN: 5.33%

Parting thoughts

- The AI model, with and without physics constraints, can successfully generate 3D realizations of the time-averaged flow field of MHK turbine arrays in large-scale meandering rivers.

- The physics constraints significantly reduce the divergence and momentum indices in the predicted flow field without impacting the computational cost or the accuracy of the predictions.

- The developed AI generates the time-averaged flow field using roughly 2.8% of the LES computational cost.

Questions