Transfer Learning in Multi-fidelity Surrogate Modeling : A Wind Farm Case

Anonymous Authors[1]

Abstract

Multi-fidelity surrogate modeling aims to describe complex systems governed by partial differential equations with few high-fidelity data points and abundant low-fidelity data points. Recent works leverage deep neural networks and few-shot transfer learning to achieve good results on several high-dimensional surrogate modeling problems. However, these works treat "multi-fidelity" as "multi-resolution" where low-fidelity simulations are computed using the same algorithm as high-fidelity simulations but with coarser grids. In real practice, low-fidelity simulations are often computed by approximating hard-to-compute terms and neglecting physics that are difficult to model. The features learned from low-fidelity data are not useful for predicting phenomena caused by those ignored physics. During fine-tuning, new features that the model learns for these regions will be inaccurate and can corrupt the pre-trained features. This can create unnecessary uncertainty for the predictions of regions that are less dependent on ignored physics. To overcome this problem, we propose a multi-step transfer learning method that, in each step, adaptively relaxes the constraint on model weights and collects regional pseudo-high-fidelity data to enlarge the training set. Our experiments on modeling wind farm flow fields show that our method significantly outperforms vanilla transfer learning methods.

1. Introduction

Recently, data-driven neural operator learning has achieved great success in building surrogate models for many computational problems such as weather forecasting (Pathak et al., 2022), seismic wave propagation (Yang et al., 2021), and CO_2 migration (Wen et al., 2022). However, producing enough data for the training of deep neural networks (DNNs) poses a challenge when the solution of the partial differential equations (PDEs) that describe our systems is computationally expensive. Alternatively, low-fidelity models can offer reasonably accurate solutions at a lower computational cost. Therefore, training DNN models with a combination of low- and high-fidelity data may offer a solution when the quantity of the latter is limited. Previous works (Chen & Stinis, 2024; De et al., 2020; Zhang et al., 2023; Lyu et al., 2023; Liao et al., 2021) have used the idea of few-shot transfer learning to the multi-fidelity modeling of high or infinite dimensional outputs with deep neural operators, achieving good results. However, these works treat "multi-fidelity" as "multi-resolution", where low-fidelity simulations are computed using the same algorithm as in the high-fidelity simulations but on coarser grids.

Low-fidelity simulations can also come from simplified model compared to high-fidelity simulations. These often neglect or approximate some hard-to-compute terms found in high-fidelity models. Consequently, they can yield significant speed improvement compared to merely employing coarser grids. For example, the Euler method ignores high-order terms and only keeps first-order terms during the propagation of ordinary differential equations (ODEs). Another example is Reynolds-Averaged Navier-Stokes (RANS) (Reynolds, 1895), which computes the mean flow by modeling the effect of the turbulence fluctuations by approximating the Reynolds stresses. The ignored or approximated terms have different effects at different regions of the output space. The Euler method will have higher errors as time increases. The errors between RANS and high-fidelity simulations show spatial dependence as shown in previous works (Rumsey & Nishino, 2011; Breuer et al., 2003; Rodi, 1997).

While previous works successfully apply vanilla transfer learning to the first type of multi-fidelity, we found that directly applying vanilla transfer learning methods to the second type of multi-fidelity yields inaccurate results. The relationship between the pre-trained features and high-fidelity data varies significantly across different regions of the simulation. We hypothesize that this is the reason for low performance. For the regions that are largely dependent on the physics neglected by low-fidelity data, the pre-trained features are less related to high-fidelity simulations. During

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review at ICML 2024 AI for Science workshop. Do not distribute.

fine-tuning, the model will learn new features from these regions and corrupt the pre-trained features. However, for the regions where observed high-fidelity simulations largely match the physics preserved in the low-fidelity data, the pre-trained features are useful for making accurate predictions. The corrupted pre-trained features can degrade the prediction accuracy in these regions.

We propose a multi-step transfer learning method to deal with this problem based on a multi-task network. By increasing the weight on high-fidelity outputs and reducing the weight on low-fidelity outputs in the loss function, we gradually increase the flexibility of the model to first fit highly related regions and then to less related regions. We collect pseudo high-fidelity data on the way of reducing constraint so that when we increase the model flexibility, the uncertainty of already well-fitted regions will not increase.

Since the second type of multi-fidelity has not been studied before in high-dimensional surrogate modeling, there are no datasets that can be used to evaluate model performance on the second type of multi-fidelity. We create one test case of wind farm mean flow predictions and evaluate our method on this problem. Our experiments show that our method significantly outperforms the vanilla transfer learning method. We hope to collect more test cases in the future and create a benchmark dataset for second-type multi-fidelity modeling.

The main contributions of our paper are:

- We point out the second type of multi-fidelity, which is generally ignored in recent works, and show that methods successful in the first type of multi-fidelity are not directly applicable to the second type of multi-fidelity.
- We propose a multi-step transfer learning method to solve the second type of multi-fidelity surrogate modeling problem.
- We construct a test case of wind farm mean flow predictions and design a network that can efficiently predict the wind farm mean flow with input from the input parameter space. Empirical results show that the vanilla transfer learning method is not fit for the second type of multi-fidelity problems and our method significantly outperforms the vanilla baseline.

2. Background

Multi-fidelity surrogate modeling aims to build models emulating high-fidelity simulations with data from different fidelity levels, where low-fidelity data are often more abundant compared to high-fidelity data. Multi-Fidelity Kriging (MFK) (Kennedy & O'Hagan, 1998; Kennedy & O'Hagan, 2001) and its variants (Damianou & Lawrence, 2013; Le Gratiet, 2012; Perdikaris et al., 2015) use an autoregressive model to integrate simulations of different fidelity levels with a Gaussian Process. MFK has been the standard solution for low-dimensional multi-fidelity problems. Forrester (Forrester et al., 2007) is the most classic test function for these methods. Surjanovic & Bingham also proposes some common functions used to evaluate multi-fidelity models, including Borehole, Currin, and Park91 A and B functions. However, their outputs are only one-dimensional scalar values. Therefore, they are not good test functions for high-dimensional neural operator methods. MFK is also not scalable for high-dimensional data.

Few-shot transfer learning aims to adapt a model pre-trained on large-scale datasets for a downstream task with a limited amount of data. Such adaptation is often realized by fine-tuning. Intuitively, the fine-tuned model will have better performance if the pre-trained features are more related to the downstream task. Zhou et al. (2021) gives an upper bound on the test error of fine-tuning empirical risk minimizer (ERM) which depends on the L_2-distance between pre-trained model weights and fine-tuned model weights. Hu et al. (2024b;a) explicitly define a model-agnostic method to calculate "task distance" as a measurement for task similarity in classification problems. Zamir et al. (2018) and later works (Dwivedi & Roig, 2019; Sun et al., 2019; Liu et al., 2020) consider heterogeneous similarities in the pre-train task. They try to divide pre-training tasks into different subsets of tasks based on task similarity and choose the best subset of pre-training sub-tasks for different downstream tasks. These methods consider the heterogeneous similarity in pre-training tasks and focus on building a better pre-trained model by choosing more related pre-training sub-tasks and abandoning less related sub-tasks. They consider the downstream task as a whole and compare it to different subsets of pre-training tasks. Our paper considers the similarity of the physics underlying the high and low fidelity simulations at different regions of the simulation for the *same* task and proposes a better fine-tuning strategy.

Multi-fidelity surrogate modeling through transfer learning pre-trains a deep neural network on low-fidelity data and fine-tunes the network on high-fidelity data. Deep neural networks like DeepONet (Lu et al., 2019) and FNO (Li et al., 2020) have shown great ability as data-driven surrogate models for high- or infinite-dimensional outputs. The computational cost of collecting enough training data for neural-network training urges us to adapt these data-driven models to multi-fidelity surrogate modeling. Transfer learning is a natural approach and has been studied in different problems (Chen & Stinis, 2024; De et al., 2020; Zhang et al., 2023; Lyu et al., 2023; Liao et al., 2021). Except for passive learning, multi-fidelity active learning (Li et al., 2022a; Wu et al., 2023; Li et al., 2022b) tries to balance between information gain and computational cost and actively decide

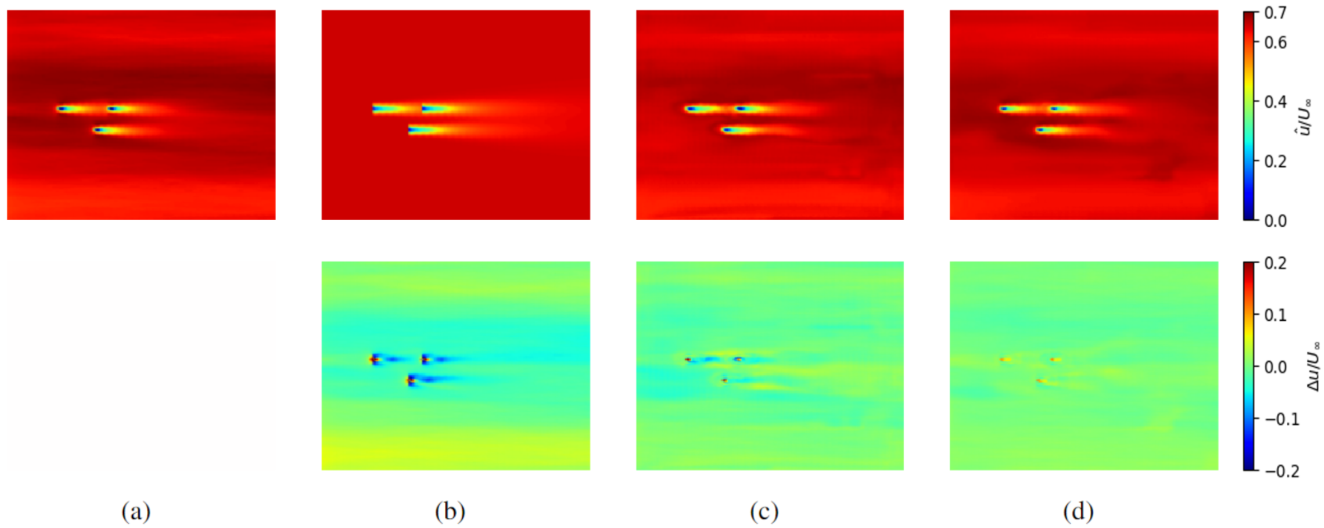


Figure 1. Predicted flow of SWiFT wind farm under 13m/s Northeast wind. Each column visualizes a slice at turbine hub height parallel to the x-y plane of the predicted flow (first row) and the error between this prediction and the high-fidelity simulation (second row). (a) High-fidelity simulations (b) Low-fidelity simulations (c) model w/ vanilla transfer learning (d) our model.

the fidelity level of the next data to acquire. However, all of these works treat "multi-fidelity" as "multi-resolution" for experiments with high-dimensional outputs and collect data from the same algorithms with different resolutions of grids.

Semi-supervised learning combines a small amount of labeled data with a large amount of unlabeled data to improve learning accuracy. Pseudo labeling (Lee, 2013) assigns pseudo labels to unlabeled data points based on the predictions of a model. It enforces the classifiers to make confident predictions on the unlabeled data and reduce prediction uncertainty. Yalniz et al. (2019) works on a semi-supervised learning scenario similar to multi-fidelity surrogate modeling where there are abundant images with low-fidelity labels (tags) and few images with high-fidelity labels (class). They pre-train their model first on low-fidelity labels and then fine-tune the model with semi-supervised methods. However, this multi-fidelity problem is still low-dimensional. The pseudo-label-selecting method in image classification is also not applicable in collecting regional pseudo data in high dimensional continuous functional space.

Wake flows refer to the movement of fluid streams interacting with objects along their path. Accurate simulations of mean wake flows are important in many areas, such as aircraft and vehicle design and wind farm optimization. While computational fluid dynamics methods like direct numerical simulations (DNS) and large eddy simulations (LES) (Zhiyin, 2015) offer high-fidelity predictions, they come with inherent challenges. DNS requires a grid resolution smaller than the smallest dynamically significant length scale (the Kolmogorov micro-scale), which makes it prohibitively expensive. Although LES requires a lower grid resolution than DNS, it still requires a large number of computational grid nodes to resolve vortices in turbulent flow, so it is computationally expensive, especially in scenarios like large-scale wind farms.

Reynolds-averaged Navier Stokes (RANS) (Reynolds, 1895) offers another numerical prediction method of time-averaged flow with less accuracy and lower cost compared to LES. It directly models the time-averaged flows by modeling fluctuations with some approximation of the Reynolds stress. The Gaussian wake model (GWM) (Niayifar & Porté-Agel, 2016; Bastankhah & Porté-Agel, 2016) is an even simpler analytical model for time-averaged wind farm flow field prediction. It assumes the velocity deficit of the flow follows Gaussian distributions in the spanwise direction and resolves the whole velocity fields based on the conservation of mass and momentum.

Mean wake flows are good testing cases for high-dimensional surrogate modeling. For every simulation of an entire flow field, fluid acts differently at different regions (Th.Yao-Tsu, 2005) from free stream overhead to extremely chaotic behavior when interacting with objects and finally goes to gradually steady far-wake. The approximations made by RANS and GWM have different effects at different regions and create heterogeneous similarities to LES. Wind farm flow fields have even more diverse behavior since they involve the wake of turbines ahead and interacting with turbines behind. Fig 1(a)(b) shows a comparison between LES and GWM of SWiFT wind farm.

3. Multi-step Transfer Learning

Since fitting the model to some regions can corrupt the pre-trained features, creating unnecessary uncertainty for other regions highly correlated with the pre-trained features, we should constrain the model from drifting too far away from the pre-trained one. However, strong constraints will also limit the ability of the model to learn new features for regions with low correlation with pre-trained features, creating prediction bias in these regions. To overcome the bias and variance problems in different regions simultaneously, instead of using a fixed constraint, we use a constraint that can be adjusted by some control parameters γ, where $\gamma = 1$ represents the strongest constraint and $\gamma = 0$ represents no constraint. We gradually release γ from 1 to 0. For each value of γ, we train the model until optimized. We sample data points from the parameter space, evaluate the regional similarities between high- and low-fidelity simulations at these data points, and select some parts of them as pseudo-high-fidelity data. The idea of collecting these pseudo data is to reduce the uncertainty of model predictions for regions already well-fitted when we further reduce the constraint.

We implement model constraint and pseudo-data selection with a multi-task network. Let $S_L = (x_i, y_i^L)$ be the low-fidelity dataset, $S_H = (x_j, y_j^H)$ be the high-fidelity dataset, f_θ be some neural backbone, A_L and A_H be two linear layers. $\hat{y}_i^L = A_L f_\theta(x_i)$ are the low-fidelity predictions and $\hat{y}_j^H = A_H f_\theta(x_j)$ are the high-fidelity predictions. The network is trained to minimize a multi-task loss:

$$\mathcal{L} = \frac{\gamma}{|S_L|}\sum_i ||\hat{y}_i^L - y_i^L||_2 + \frac{(1-\gamma)}{|S_H|}\sum_j ||\hat{y}_j^H - y_j^H||_2.$$

Here, we analyze how the value of γ determines the constraint on the model. When $\gamma = 1$, the model will be trained to match low-fidelity simulations. When $\gamma = 0$, the model will be trained to match high-fidelity simulations. Taking only the 1 and 0 values of γ without pseudo labeling will be a vanilla transfer learning process. For $\gamma \to 1$, the model learns to solve a constrained optimization problem to minimize $||\hat{y}^H - y^H||_2$ subject to $\hat{y}^H \approx A_L A_H^* y^L$ where A_H^* is the pseudo-inverse of A_H. Therefore $\gamma \to 1$ is the same as the constraint that the predicted high-fidelity outputs must be approximately a linear transformation of the low-fidelity simulations. For other values of γ, this multi-task loss generally constrains how far away the high-fidelity predictions can be from a subspace spanned by some linear transformations of their corresponding low-fidelity ones.

To select regional pseudo data, we partition high-dimensional outputs into regions. Let V be the output space, $P(V) = \{v_k\}$ be some partition over V and $n(v_k)$ be the relative volume of v_k with respect to V. For some regions v_k and some functional similarity measure $\mathcal{F}$, the regional similarity is $\frac{1}{n(v_k)}\int_{v_k}\mathcal{F}(y^L, y^H)dv$. However, since we do not have y^H for x we use the estimated one as $\hat{y}^H = A_H f_\theta(x)$ and the estimated regional similarity is $\frac{1}{n(v_k)}\int_{v_k}\mathcal{F}(A_H f_\theta(x), y^H)dv$. Fig 7 shows the ground truth and estimated regional similarity in the wind farm experiment with $\gamma = 0.5$ and L^2-norm as similarity measurement. The estimated similarities approximate the ground truth well. It also shows that measured similarity values lie in distinct groups. This matches our assumption that similarities are varied at different regions due to the amount of physics ignored by low-fidelity methods.

Algorithm 1 Multi-step transfer learning

Input: low-fidelity dataset $S_L = \{(x_i, y_i^L)\}$; high-fidelity datset $S_H = \{(x_j, y_j^H, V)\}$; neural backbone f_θ; Linear layers A_L and A_H; Multi-task loss function $\mathcal{L}$; similarity measurement $\mathcal{F}$; output space partition $P(V)$; threshold parameter δ, η

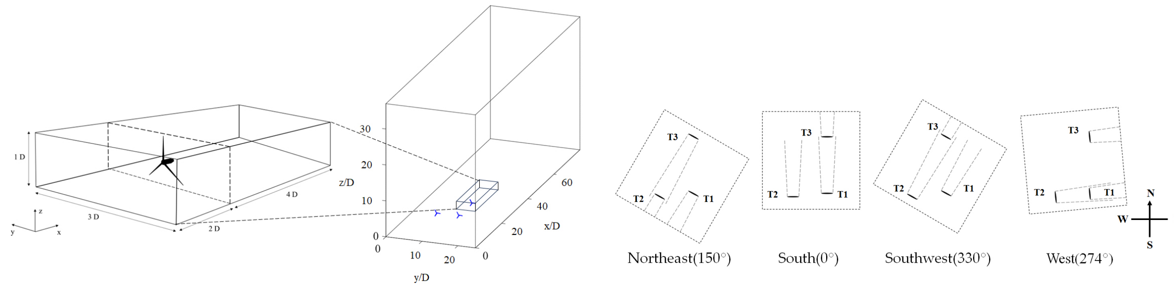
```
for γ = 1 to 0 do
  Minimize L
  if γ ≠ 1 then
    for (x, y^L) in S_L do
      for v in P(V) do
        if 1/n(v) ∫_v F(A_H f_θ(x), y^L) dv < 1/((δ+γ)η) then
        Add (x, A_H f_θ(x), v) to S_H
        end if
      end for
    end for
  end if
end for
```

As we gradually increase flexibility, we want to collect pseudo data at the regions where high- and low-fidelity simulations are less relevant. Therefore, We use an adaptive threshold $\frac{1}{(\delta+\gamma)\eta}$, where γ is the constraint control parameter and δ and η are some pre-selected threshold parameters. If the estimated similarity is smaller than this threshold at region v of input x, we add $(x, A_H f_\theta(x), v)$ to the high-fidelity training set.

After regional pseudo data are added to the training set. Not all data points in high-fidelity datasets have ground truth over the entire output space V. We represent the new high-fidelity training set as $S_H = \{(x_j, y_j^H, v_j)\}$, where v_j is the confident regions in V for input x. $v_j = V$ if the data point is from the original training set. The multi-task loss will be modified as:

$$\mathcal{L} = \frac{\gamma}{|S_L|}\sum_i ||\hat{y}_i^L - y_i^L||_2 + \frac{(1-\gamma)}{\sum_j n(v_j)}\sum_j ||(\hat{y}_j^H - y_j^H)\mathbf{1}_{v_j}||_2,$$

where $\mathbf{1}_{(\cdot)}$ represents the indicator function of confident regions.



(a) (b)

Figure 2. (a) Illustration of the South wind direction LES configuration and a zoomed-in region surrounding a turbine. This turbine is located closely downwind to another turbine, and therefore, it is affected by the upwind turbine wake. (b) Illustration of the simulation box under different wind directions. The simulation box is adjusted so that the x-axis is always along the current wind direction.

4. Wind Farm Wake Flow Prediction

4.1. Dataset

We collect high-fidelity data by using LES to simulate the Sandia National Laboratories Scaled Wind Farm Technology (SWiFT) site (Berg et al.) in Lubbock, Texas, which includes three Vestas V27 turbines. The wind turbine layout is shown in Fig 2(a). The wind turbines, Vestas V27s, have rotor diameters of D = 27m and hub heights of 32.1m. These simulations have 1.8×10^7 grid nodes and took 8×10^4 CPU hours to converge. The amount of data collected was strictly limited by the high computational cost. We run LES under five different wind speeds (bulk speed U_∞ = 7, 9, 11, 13, 15 m/s) and four different wind directions (150°, 0°, 330°, and 274°, taking south as 0°). We denote these wind directions as Northeast, South, Southwest, and West (Fig 2(b)). They were selected so that there was always one turbine directly downwind of another turbine and therefore was affected by the wake of the upwind turbine. This ensures that our model is trained and tested on "hard examples". For all wind directions, wind turbines are adjusted to directly facing the up-coming wind (0° yaw angles). We calculate the time-average flow fields by averaging LES results until they statistically converge.

We choose GWM to generate low-fidelity data. GWM is much faster than RANS and we can do dense sampling with it. GWM is also less accurate compared to RANS. This makes GWM-LES a harder multi-fidelity problem compared to RANS-LES. We used FLOw Redirection and Induction in Steady State (FLORIS) to generate the low-fidelity simulations of the same site. We densely sampled the wind directions from 0° to 359° with a spacing of 1° and the wind speeds from $7m/s$ to $15m/s$ with a spacing of $1m/s$. The generation of low-fidelity simulations requires less than one minute per case. This creates a total of 3,240 low-fidelity cases.

We construct the training set with three high-fidelity cases (Southwest 7m/s, South 11m/s, and West 15m/s) and all low-fidelity cases. The other high-fidelity simulations are test cases. Specifically, simulations of Northeast wind directions do not exist in the training set. They are considered to be out-of-distribution test cases to evaluate the model's ability to extrapolate/generalize.

4.2. Model Architecture

We separate parameters into two groups: one describing the wind condition and the other describing the wind farm condition - and process them separately.

We use the wind condition to approximate the free-stream flow field by the law of the wall (Kármán, 1930). The law of the wall states that the velocity of the fluid at a point in the boundary layer depends logarithmically on the distance from the wall. This can be viewed as an approximation of wind farm flow without any turbine. Although it is a coarse approximation, it can be calculated efficiently on GPUs and produces a representation that contains all the wind information and is suitable for the convolutional neural network (CNN).

We encode the layout of the wind farm by representing each turbine as a 3-D Gaussian distribution in the wind farm whose mean is at the turbine center location and variance is determined by the covariance matrix:

$$\begin{bmatrix} (4cos^2(\theta)+sin^2(\theta))D^2 & 2sin(2\theta)D^2 & 0 \\ 2sin(2\theta)D^2 & (4sin^2(\theta)+cos^2(\theta))D^2 & 0 \\ 0 & 0 & D^2 \end{bmatrix}$$

where D is the turbine diameter and θ is the blade yaw angle of the turbine. Representing object locations as Gaussian distributions is a common technique in computer vision. Moreover, as (Niayifar & Porté-Agel, 2016; Bastankhah & Porté-Agel, 2016) has shown that turbine wake deficit is approximately Gaussian in the spanwise direction, we also believe that this 3-D Gaussian representation of wind farm

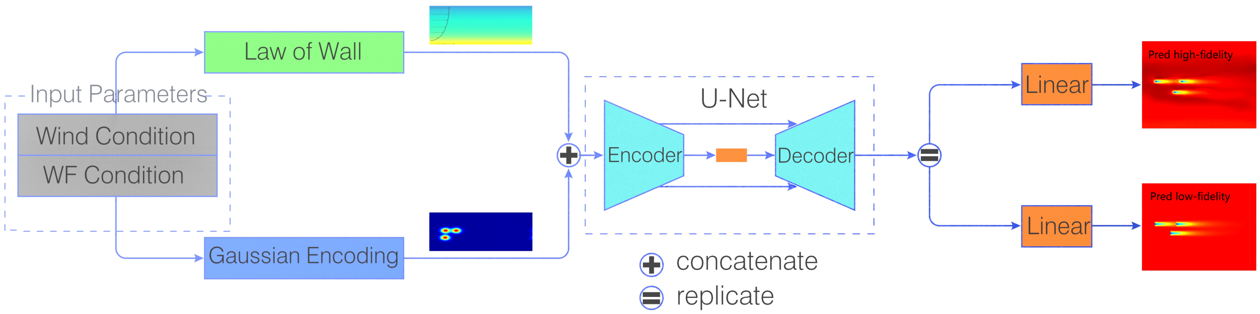


Figure 3. Model Architecture. Parameters are pre-processed separately. Parameters describing the upcoming wind are encoded by the law of wall. Parameters describing the wind-farm layout are encoded by 3-D Gaussian. The encoded representations are concatenated together. The concatenated representation is passed to a U-Net backbone. Two linear layers take U-Net output and produce predictions of GWM and LES.

to the actual flow field is an easy-to-learn mapping.

We concatenate both representations and pass them to a U-Net (Ronneberger et al., 2015) backbone. Two linear layers are used for multi-task outputs for GWM and LES predictions. Fig 3 illustrates the architecture of our model.

4.3. Collecting Pseudo data

Previous work on semi-supervised learning of high-dimensional outputs (Li et al., 2023) uses patch-level pseudo labels. We follow this convention and use patch-level pseudo data in our problem as regional pseudo data. Rather than evenly dividing the parameter space into patches, we define finer patches around wind turbines. For each turbine, we define three patches. All of them are $3D$ wide in the y-axis and $1D$ tall in the z-axis, where D is the turbine blade diameter. In the wind flow direction, x-axis, the patches are divided based on their distance to the turbine center. These patches include overhead patch ($2D$ to $0.5D$ ahead), turbulence patch ($0.5D$ ahead to $1D$ behind), wake patch ($1D$ behind to $10D$ behind). Furthermore, if a patch A of a turbine intersects with patch B of another turbine, we divide the patch into A/B, B/A, and $A \cap B$. After we create turbine-related patches, the other background patches are evenly divided. Each of them is a cube with (Length, Width, Height) = $\frac{1}{5}$ (Length, Width, Height) of the whole simulation box. For background patch A and turbine-related patch B, if $A \cap B \neq \emptyset$, reassign A as A/B. This ensures no overlap patches.

4.4. Training

We take $\gamma = 1, 0.5, 0$, $\eta = 50$, $\delta = 0$ and use L^2-norm as similarity measurement. For each value of γ, the network is trained for 100,000 iterations using the Adam optimizer (Kingma & Ba, 2014) with a learning rate of 10^{-4}. For each iteration, we choose a mini-batch of 20 low-fidelity data points and 20 high-fidelity data points.

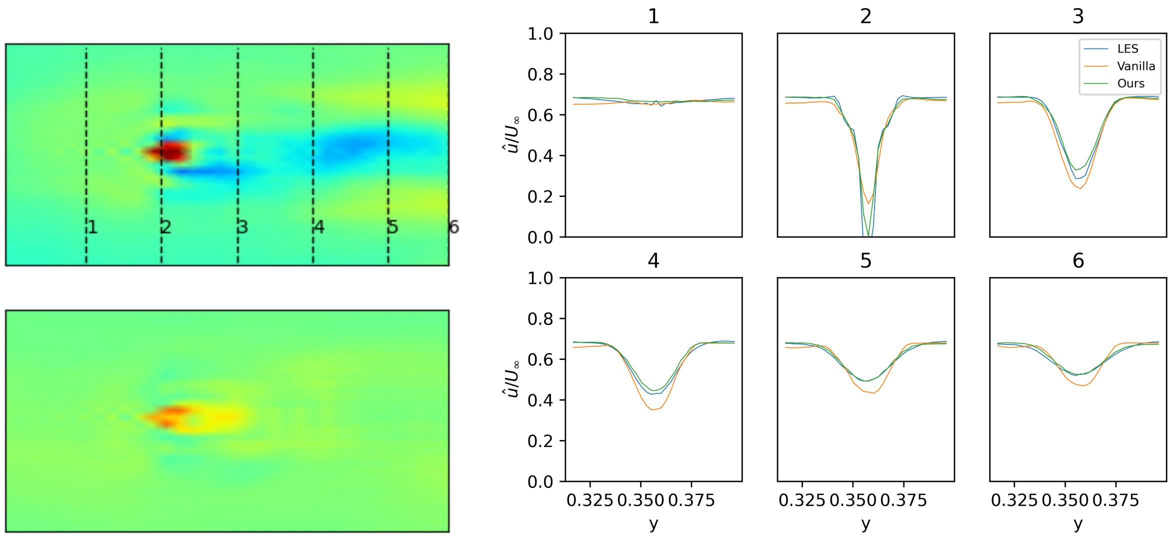
The vanilla transfer learning baseline is trained with exactly the same setup except that γ only takes two values: 0 and 1, and there is no pseudo labeling.

4.5. Results

We evaluate the model performance by root mean squared error (RMSE) between model-predicted flows and high-fidelity flows as shown in Tab 1. Since the entire simulation box is dominated by free-stream, we report the RMSE between model-predicted flows and high-fidelity flows only at zoomed-in regions around turbines. Such zoomed-in regions are defined as in Fig 2(a). In all wind directions, our method outperforms vanilla transfer learning methods.

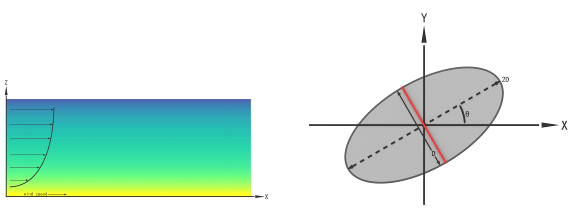
We also plot pixel-level 2-D histograms of model-predicted values against high-fidelity values in Fig 6. Prediction errors of our model are significantly less variate compared to predictions from the vanilla model across different regions. This demonstrates that our model successfully reduces prediction uncertainty. Predictions of our model have some bias at a few places. The reason may be that we accidentally collected regional pseudo data that are not yet well fitted. However, since the reduced uncertainty is much more significant than the induced bias, our model still outperforms the vanilla one by a large margin.

High-fidelity flow fields of Northeast wind direction are not present in the training cases. Therefore, these are considered out-of-distribution testing cases to evaluate the model extrapolation/generalization ability. The predicted fields of the time-averaged velocity at the hub-height plane of the most Northeast 13 m/s case are shown in Fig 1. We also provide a detailed comparison of the mean flow prediction



(a) (b)

Figure 4. Visualization of region surrounding turbine 3 in Fig 1. (a) Errors between high-fidelity ground truth and prediction by vanilla model (first row) and Errors between high-fidelity ground truth and prediction by our model (second row). Our model has better performance in all regions. (b) Time-averaged velocity profiles along the spanwise direction. Profiles are taken along the y-direction as denoted by the dashed lines on (a) as labeled 1 through 6. For each profiles, we show predicted relative velocities from LES(-), vanilla model(-), and our model(-). Predicted velocities of our model closely agree with LES and is much better than velocities calculated from vanilla model.



(a) (b)

Figure 5. (a) Illustration of the free stream velocity field constructed using the law of the wall. Velocities are increasing logarithmically along z-axis (Height). (b) Illustration of Gaussian representation corresponding to a turbine; Red line represents the turbine blades; 1-σ area of the Gaussian distribution is painted gray. Both figures have the same coordinates as in Fig 2(a) with x-axis as the flow direction, y-axis as spanwise direction, and z-axis as vertical direction.

error maps and velocity profiles around a specific turbine in Fig 4. Our model significantly outperforms the vanilla one in all regions.

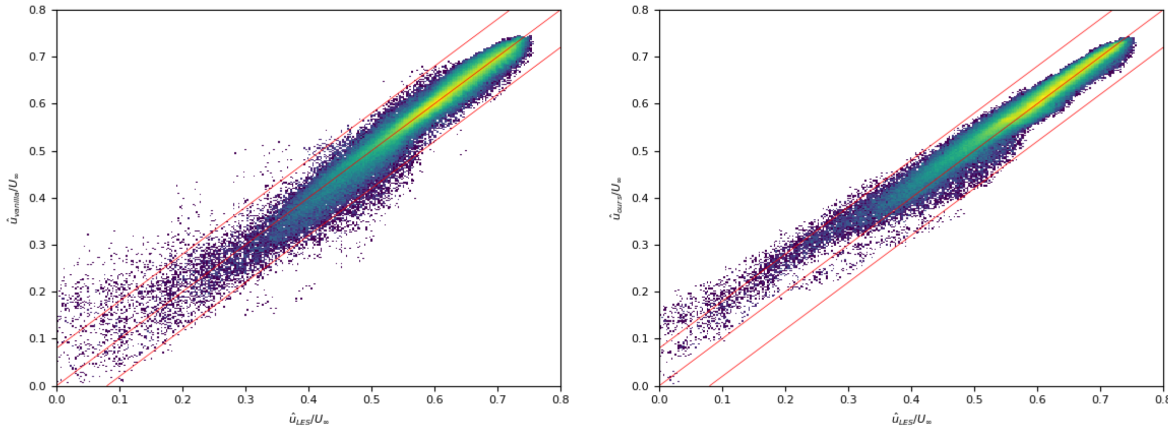
5. Discussion and Future Works

Multi-fidelity learning is a potential solution for data-driven neural operator learning if the computational cost of collecting high-fidelity simulations is prohibitively high. While multiple recent works are focusing on this problem, there is no standard benchmark dataset for the evaluation of high-dimensional multi-fidelity surrogate modeling. The datasets they generated on their own simply treat "multi-fidelity" as "multi-resolution".

Table 1. RMSE of the zoomed-in regions of all turbines under different wind directions, averaged over different wind speeds. Vanilla: model trained with vanilla transfer learning; Multi-step: model trained with our method.

WIND DIRECTION	VANILLA	MULTI-STEP
WEST	0.0188	0.0185
SOUTH	0.0165	0.0144
SOUTHWEST	0.0179	0.0157
NORTHEAST	0.0195	0.0159

In this work, we argue that models that work well on "multi-resolution" learning may not generalize to multi-fidelity learning problems where low-fidelity data is simulated by ignoring or approximating hard-to-compute terms. We create a test case of wind farm mean flow prediction to evaluate model performance on the second type of multi-fidelity. We propose a multi-step transfer learning method that shows better performance on wind farm mean flow predictions than



(a) (b)

Figure 6. Pixel-level 2D histograms of high-fidelity relative velocities against model-predicted relative velocities. Pixels are collected from the zoomed-in regions of all turbines under different wind directions and different wind speeds. (a) Histogram of vanilla-model predictions, having high variance across all values; (b) Histogram of our-model predictions, significantly reducing the variance.

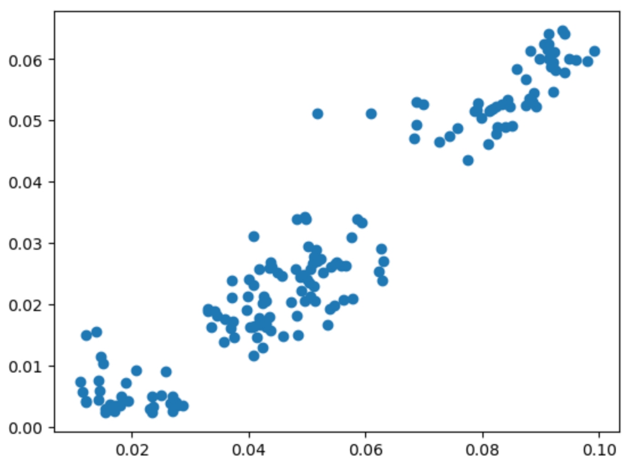


Figure 7. Scatter plot of estimated similarities (y-axis) and actual similarities (x-aixs) of different patches with $\gamma = 0.5$ and L^2-norm as similarity measure. Values are evaluated at the data points where LES is available but not exist in training set.

vanilla transfer learning. However, a single test case is not enough. We hope to collect more test cases that represent both the first and second types of multi-fidelity and create a benchmark dataset for high-dimensional multi-fidelity surrogate modeling in the future.

Our works also contribute to the wind energy management community. Based on our method, we build the first end-to-end wind-farm mean-flow prediction model that requires a very small amount of high-fidelity supervision, achieves high accuracy, and is trainable and can perform inference efficiently on a single commercial GPU. We also plan to test its engineering applicability for real-world wind-farm design, analysis, and control in the future.

References

Bastankhah, M. and Porté-Agel, F. Experimental and theoretical study of wind turbine wakes in yawed conditions. *Journal of Fluid Mechanics*, 806:506–541, 2016.

Berg, J., Bryant, J., LeBlanc, B., Maniaci, D. C., Naughton, B., Paquette, J. A., Resor, B. R., White, J., and Kroeker, D. *Scaled Wind Farm Technology Facility Overview*.

Breuer, M., Jovičić, N., and Mazaev, K. Comparison of des, rans and les for the separated flow around a flat plate at high incidence. *International journal for numerical methods in fluids*, 41(4):357–388, 2003.

Chen, W. and Stinis, P. Feature-adjacent multi-fidelity physics-informed machine learning for partial differential equations. *Journal of Computational Physics*, 498: 112683, 2024.

Damianou, A. and Lawrence, N. D. Deep Gaussian processes. In Carvalho, C. M. and Ravikumar, P. (eds.), *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, volume 31 of *Proceedings of Machine Learning Research*, pp. 207–215, Scottsdale, Arizona, USA, 29 Apr–01 May 2013. PMLR.

De, S., Britton, J., Reynolds, M., Skinner, R., Jansen, K., and Doostan, A. On transfer learning of neural networks using bi-fidelity data for uncertainty propagation. *International Journal for Uncertainty Quantification*, 10(6), 2020.

Dwivedi, K. and Roig, G. Representation similarity analysis for efficient task taxonomy & transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12387–12396, 2019.

FLORIS. Floris. version x.y.z (2018). available at https://github.com/wisdem/floris.

Forrester, A., Sobester, A., and Keane, A. Multi-fidelity optimization via surrogate modelling. *Proc. R. Soc. A*, 463:3251–3269, 10 2007. doi: 10.1098/rspa.2007.1900.

Hu, M., Chang, H., Guo, Z., Ma, B., Shan, S., and Chen, X. Task attribute distance for few-shot learning: Theoretical analysis and applications, 2024a.

Hu, M., Chang, H., Guo, Z., Ma, B., Shan, S., and Chen, X. Understanding few-shot learning: Measuring task relatedness and adaptation difficulty via attributes. *Advances in Neural Information Processing Systems*, 36, 2024b.

Kennedy, M. and O'Hagan, A. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87, 10 1998. doi: 10.1093/biomet/87.1.1.

Kennedy, M. C. and O'Hagan, A. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63, 2001. URL https://api.semanticscholar.org/CorpusID:119562136.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

Kármán, T. v. Mechanische aenlichkeit und turbulenz. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, 1930:58–76, 1930. URL http://eudml.org/doc/59299.

Le Gratiet, L. Recursive co-kriging model for design of computer experiments with multiple levels of fidelity. *International Journal for Uncertainty Quantification*, 4, 10 2012. doi: 10.1615/Int.J.UncertaintyQuantification.2014006914.

Lee, D.-H. Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. 2013. URL https://api.semanticscholar.org/CorpusID:18507866.

Li, C., Hu, X., Abousamra, S., and Chen, C. Calibrating uncertainty for semi-supervised crowd counting. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16685–16695. IEEE, 2023.

Li, S., Phillips, J. M., Yu, X., Kirby, R., and Zhe, S. Batch multi-fidelity active learning with budget constraints. *Advances in Neural Information Processing Systems*, 35:995–1007, 2022a.

Li, S., Wang, Z., Kirby, R., and Zhe, S. Deep multi-fidelity active learning of high-dimensional outputs. In Camps-Valls, G., Ruiz, F. J. R., and Valera, I. (eds.), *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 1694–1711. PMLR, 28–30 Mar 2022b.

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

Liao, P., Song, W., Du, P., and Zhao, H. Multi-fidelity convolutional neural network surrogate model for aerodynamic optimization based on transfer learning. *Physics of Fluids*, 33, 12 2021. doi: 10.1063/5.0076538.

Liu, C., Wang, Z., Sahoo, D., Fang, Y., Zhang, K., and Hoi, S. C. Adaptive task sampling for meta-learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pp. 752–769. Springer, 2020.

Lu, L., Jin, P., and Karniadakis, G. E. Deeponet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. *arXiv preprint arXiv:1910.03193*, 2019.

Lyu, Y., Zhao, X., Gong, Z., Kang, X., and Yao, W. Multi-fidelity prediction of fluid flow and temperature field based on transfer learning using fourier neural operator. *arXiv preprint arXiv:2304.06972*, 2023.

Niayifar, A. and Porté-Agel, F. Analytical modeling of wind farms: A new approach for power prediction. *Energies*, 9 (9), 2016.

Pathak, J., Subramanian, S., Harrington, P., Raja, S., Chattopadhyay, A., Mardani, M., Kurth, T., Hall, D., Li, Z., Azizzadenesheli, K., Hassanzadeh, P., Kashinath, K., and Anandkumar, A. Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators. *arXiv preprint arXiv:2202.11214*, 2022.

Perdikaris, P., Venturi, D., Royset, J. O., and Karniadakis, G. E. Multi-fidelity modelling via recursive co-kriging and gaussian–markov random fields. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2179):20150018, 2015.

Reynolds, O. On the Dynamical Theory of Incompressible Viscous Fluids and the Determination of the Criterion. *Philosophical Transactions of the Royal Society of London Series A*, 186:123–164, January 1895. doi: 10.1098/rsta.1895.0004.

Rodi, W. Comparison of les and rans calculations of the flow around bluff bodies. *Journal of wind engineering and industrial aerodynamics*, 69:55–75, 1997.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. F. (eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pp. 234–241, Cham, 2015. Springer International Publishing.

Rumsey, C. L. and Nishino, T. Numerical study comparing rans and les approaches on a circulation control airfoil. *International Journal of Heat and Fluid Flow*, 32(5):847–864, 2011. ISSN 0142-727X. doi: https://doi.org/10.1016/j.ijheatfluidflow.2011.06.011. URL https://www.sciencedirect.com/science/article/pii/S0142727X11001007.

Sun, Q., Liu, Y., Chua, T.-S., and Schiele, B. Meta-transfer learning for few-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 403–412, 2019.

Surjanovic, S. and Bingham, D. Virtual library of simulation experiments: Test functions and datasets. Retrieved May 20, 2024, from http://www.sfu.ca/~ssurjano.

Th.Yao-Tsu, W. Cavity and wake flows. 2005. URL https://api.semanticscholar.org/CorpusID:14947157.

Wen, G., Li, Z., Azizzadenesheli, K., Anandkumar, A., and Benson, S. M. U-fno—an enhanced fourier neural operator-based deep-learning model for multiphase flow. *Advances in Water Resources*, 163:104180, 2022.

Wu, D., Niu, R., Chinazzi, M., Ma, Y., and Yu, R. Disentangled multi-fidelity deep bayesian active learning. In *International Conference on Machine Learning*, pp. 37624–37634. PMLR, 2023.

Yalniz, I. Z., Jégou, H., Chen, K., Paluri, M., and Mahajan, D. Billion-scale semi-supervised learning for image classification, 2019.

Yang, Y., Gao, A. F., Castellanos, J. C., Ross, Z. E., Azizzadenesheli, K., and Clayton, R. W. Seismic wave propagation and inversion with neural operators. *ArXiv*, abs/2108.05421, 2021.

Zamir, A. R., Sax, A., , Shen, W. B., Guibas, L., Malik, J., and Savarese, S. Taskonomy: Disentangling task transfer learning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018.

Zhang, Y., Gong, Z., Zhou, W., Zhao, X., Zheng, X., and Yao, W. Multi-fidelity surrogate modeling for temperature field prediction using deep convolution neural network. *Engineering Applications of Artificial Intelligence*, 123:106354, 2023.

Zhiyin, Y. Large-eddy simulation: Past, present and the future. *Chinese Journal of Aeronautics*, 28(1):11–24, 2015. ISSN 1000-9361. doi: https://doi.org/10.1016/j.cja.2014.12.007. URL https://www.sciencedirect.com/science/article/pii/S1000936114002064.

Zhou, P., Zou, Y., Yuan, X.-T., Feng, J., Xiong, C., and Hoi, S. Task similarity aware meta learning: theory-inspired improvement on MAML. In de Campos, C. and Maathuis, M. H. (eds.), *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of *Proceedings of Machine Learning Research*, pp. 23–33. PMLR, 27–30 Jul 2021.