Confidence-based Self-Corrective Learning: An Application in Height Estimation Using Satellite LiDAR and Imagery

Zhili Li[1] , **Yiqun Xie**[1*] , **Xiaowei Jia**[2]
[1]University of Maryland
[2]University of Pittsburgh
{lizhili, xie}@umd.edu, xiaowei@pitt.edu

Abstract

Widespread, and rapid, environmental transformation is underway on Earth driven by human activities. Climate shifts such as global warming have led to massive and alarming loss of ice and snow in the high-latitude regions including the Arctic, causing many natural disasters due to sea-level rise, etc. Mitigating the impacts of climate change has also become a United Nations' Sustainable Development Goal for 2030. The recent launch of the ICESat-2 satellites target on heights in the polar regions. However, the observations are only available along very narrow scan lines, leaving large no-data gaps in-between. We aim to fill the gaps by combining the height observations with high-resolution satellite imagery that have large footprints (spatial coverage). The data expansion is a challenging task as the height data are often constrained on one or a few lines per image in real applications, and the images are highly noisy for height estimation. Related work on image-based height prediction and interpolation relies on specific types of images or does not consider the highly-localized height distribution. We propose a spatial self-corrective learning framework, which explicitly uses confidence-based pseudo-interpolation, recurrent self-refinement, and truth-based correction with a regression layer to address the challenges. We carry out experiments on different landscapes in the high-latitude regions and the proposed method shows stable improvements compared to the baseline methods.

1 Introduction

Climate action is an important component of the 2030 Sustainable Development Goals (SDGs) of the United Nations [UN, 2022]. Human activities have led to widespread transformations of the Earth's environment and climate [Pörtner *et al.*, 2019]. Observational evidence suggests that the climate system at high latitudes including the Arctic is undergoing significant shifts with faster increases in air temperatures [Richter-Menge *et al.*, 2019], which have led to changes including the mass loss of Greenland Ice Sheet and glaciers, permafrost thaw, decreasing snow and ice cover on land, and increasing vegetation cover.

Satellite remote sensing systems play a key role in monitoring essential climate variables in the polar regions, which are hard to access for traditional observation. Indeed, advances in sensing instruments have continued to demonstrate promising potentials. The recent launch of ICESat-2 satellites provides a revolutionary approach to measure the surface heights at high precision to help scientists quantify and better understand the ongoing changes from a third dimension. However, in order to obtain high-resolution heights at large scale with altimeter instruments, ICESat-2 has a very narrow footprint (i.e., width of a scan line) as shown in Fig. 1(a), leaving large empty gaps between scan lines. Filling the gaps will substantially improve the representativeness of the dataset.

We aim to expand the spatial footprints of the height data by combining the single-line height observations with large-footprint satellite imagery. This height estimation task has the following challenges. First, there are often only one or a few scan lines available in each satellite image tile for the corresponding timestamp, and all height observations are spatially constrained to the single or few lines, making it difficult for traditional interpolation methods to expand spatial footprints. Second, a large proportion of pixels in satellite imagery present variations in spectral characteristics (e.g., colors) that are irrelevant to height variations, which brings challenges in building the connections between the height data and spectral imagery. Finally, an image only reflects height variations in the covered region, and the base height can be highly uncertain to set. Different landscapes may also exhibit heterogeneous patterns over space [Xie *et al.*, 2021].

While image-based height estimation has been widely studied, most existing works rely on high-contrast characteristics from man-made structures in urban areas with aerial images, which are not available in non-urban regions at large scale. Or they often require specific image types such as stereo images that are not available for most satellites [Qin, 2019]. Traditional spatial interpolation methods (e.g., Kriging) designed to fill data gaps cannot well capture complex nonlinear relationships between spectral features and heights. While their deep learning extensions (e.g., Kriging convolutional networks [Appleby *et al.*, 2020]) enhance such ability,

*Corresponding author.

their performances suffer when observations are highly localized in space. More details are discussed in Sec. 3.

We propose a spatial self-corrective learning framework to expand the coverage of surface height data in high latitudes including polar regions with the following contributions:

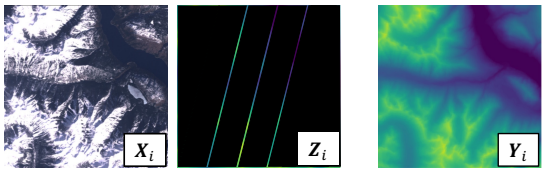
- We propose a confidence-based pseudo-interpolation framework to correct lower-confidence predictions using higher-confidence predictions from spatially-adjacent areas. This helps suppress erroneous predictions caused by irrelevant feature variations from the input imagery.
- We propose a dynamic refinement strategy with a recurrent structure to allow the pseudo-interpolation to iteratively correct low-confidence predictions, allowing adaptive self-correction during test.
- We integrate an inverse-confidence-based ridge regression layer to utilize the limited observations from the scan line (i.e., a line crossing an image) to perform truth-based correction. We further improve the robustness of the correction by using it recurrently with random ensembles. The network parameters are trained to collaborate with the recurrent regression correction to better generalize over space.
- We use base-height augmentation to reduce the overfitting effects and improve the model's usability in new test sites.

Our experiments in high-latitude regions show that the proposed self-corrective learning can stably improve height estimation compared to baseline methods on various scenarios.

2 Problem Statement

We first introduce several basic concepts for the problem.

- **Satellite footprints:** Earth observation satellites collect surface data following pre-defined orbits or scan lines. The data thus center around the scan lines. Satellites with different sensors (e.g., multispectral instrument, altimeter) cover different widths along the lines, resulting in different sizes of footprints. Large or wide footprints can cover all gaps between adjacent scan lines, whereas narrow footprints leave large empty gaps.
- **Large-footprint spectral imagery:** Satellite imagery contains multiple spectral bands (e.g., 13 bands for the Sentinel-2A product) that can range from visible (i.e., RGB) to non-visible (e.g., near-infrared) wavelengths. Spectral imagery is often collected with large footprints.
- **Large-footprint surface heights:** Surface height data can also be collected with large footprints (e.g., SRTM collected by the Space Shuttle Endeavour [Farr and Kobrick, 2000]). However, this kind of data has very limited availability at large scale or has low update frequency due to the high cost (e.g., SRTM data were last collected in 2000).
- **Narrow-footprint surface heights:** The most recent satellite constellation ICESat-2 takes a different approach by using narrow footprints at high resolution. The satellites can also continuously fly along the orbits to provide updates throughout the many-year mission. However, the narrow footprints leave large empty gaps in-between scan lines (e.g., 17m swaths vs. 3km gaps).



(a) Inputs (b) Output

Figure 1: Examples of inputs (spectral image $\mathbf{X}_i$ and narrow-footprint height $\mathbf{Z}_i$) and output (large-footprint height $\mathbf{Y}_i$). Areas in black color between scanning lines in $\mathbf{Z}_i$ are empty gaps.

- **Tile vs. patch:** Satellite images are often provided as large tiles. For example, Sentinel-2 tiles are aligned with UTM zones, where each zone is a $6^\circ \times 6^\circ$ region that can correspond to an image of size 10000×10000 ($\sim$600MBs). As such tiles are often too large as training samples for deep networks, in practice they are split into small patches (e.g., 400×400, 200×200) during preprocessing, with or without overlapping.

In the problem, we are given the following three data sources for training: (1) A collection of **large-footprint spectral imagery** tiles that are split into small patches $\mathbf{X} = \{\mathbf{X}_i \in \mathbb{R}^{m\times m\times d}\}$ where m is the patch size and d is the number of spectral bands; (2) **Narrow-footprint surface heights** along scan lines that spatially and temporally intersect with $\mathbf{X}_i \in \mathbf{X}$. We align the height data with the image patches and preprocess them into the same image patch representations $\mathbf{Z} = \{\mathbf{Z}_i \in \mathbb{R}^{m\times m\times 1}\}$, where pixels intersecting the scan lines contain the elevation values and others are set to 0 with a mask; and (3) The **large-footprint surface heights** $\{\mathbf{Y}_i \in \mathbb{R}^{m\times m\times 1}\}$ for patches $\mathbf{X}_i \in \mathbf{X}$.

The prediction problem takes $\mathbf{X}$ and $\mathbf{Z}$ as inputs and aims to predict height values that are unavailable in $\mathbf{Z}$ due to gaps in the scan lines. $\mathbf{Y}$ is used as the ground-truth labels for training. Overall, $\mathbf{Y}$ is only available for very limited image tiles in practice, so the goal is to use the trained model to expand height coverage for tiles where height is only available as $\mathbf{Z}$. Fig. 1 shows examples of the inputs and outputs.

3 Related Work

Pixel-level prediction with imagery: To derive height information at the pixel level, traditional methods mostly focus on stereo matching [Soergel *et al.*, 2009], which uses two stereo images (i.e., near-simultaneous images at different angles) to reconstruct 3D information. For example, [Liu *et al.*, 2017] estimates building heights using high-resolution stereo images in urban areas. However, stereo images are unavailable in most of the satellite platforms. Additionally, non-urban areas often do not have the high-contrast corresponding points needed by the methods. Deep semantic segmentation networks such as U-Net [Ronneberger *et al.*, 2015], DeepLabV3+ [Chen *et al.*, 2018] and so on can also be utilized for single-image-based height prediction. [Amirkolaee and Arefi, 2019] used a multi-scale encoder-decoder to estimate height using single aerial images. These methods perform well in urban areas. However, they rely on high-resolution aerial imagery and do not have the capacity to in-

corporate additional information from narrow-footprint surface heights.

Traditional interpolation: Spatial interpolation methods estimate unknown values using distance-weighted aggregation of observations. Many interpolation methods have been developed, such as inverse distance weighting [Shepard, 1968], spline [McKinley and Levine, 1998], Kriging [Cressie, 2015], etc. Among the methods, Kriging [Cressie, 2015] is the most widely adopted approach, which is a nonparametric model based on Gaussian processes. Ordinary Kriging [Wackernagel and Wackernagel, 2003] performs estimation directly using known labels (or targets), whereas universal Kriging [Caballero *et al.*, 2013] can also incorporate feature variables. However, traditional Kriging cannot model complex non-linear relationships and its performance relies on the availability of spatially-nearby observations.

Deep-learning-based interpolation: Integrations between Kriging and deep neural networks have been developed to enhance the ability to capture non-linear relationships. Kriging convolutional networks [Appleby *et al.*, 2020] approximate Kriging interpolation using a graph neural network (GNN), which enables learning information propagation between data points. Inductive Graph Neural Network Kriging (IGNNK) model [Wu *et al.*, 2021] trains a GNN to reconstruct information on random subgraph structures, allowing it to learn to generalize to unseen nodes/graphs for spatiotemporal interpolation. However, these methods are computationally expensive and they are intended for data with a smaller number of points (e.g., thousands), which is not suitable for satellite data that often have millions of pixels (points) per image. A recent work [Liu *et al.*, 2022] proposed an image-based interpolation technique that parameterizes Kriging with deep neural networks to learn single-pixel and neighborhood embeddings for interpolation. However, these methods share a common limitation with traditional Kriging, i.e., their performances degrade when points with known labels are highly localized in space.

4 Spatial Self-Corrective Learning

We decompose the spatial self-corrective learning framework into four components: (1) A confidence-based pseudo-interpolation method to correct low-confidence predictions in local neighborhoods; (2) A recurrent structure that uses the confidence-based pseudo-interpolation as a sub-routine and self-refines the height in an iterative manner; (3) An inverse-confidence-based regression layer that further corrects the predictions using limited observations (from narrow-footprint height data) in each recurrent step; and (4) A height-based augmentation to reduce overfitting.

4.1 Spatial Correction with Confidence-Based Pseudo-Interpolation

One challenge facing height prediction from spectral imagery is the high volume of irrelevant variations. For example, spectral features may change by surface properties that are not highly correlated with changes in height, such as types of soils or rocks, surface moisture, color differences between vegetation, etc. Such irrelevant variations in signals can be both smooth or sharp. They introduce additional confusion during the training process, and will likely lead to undesired or distorted variations in predicted heights. Since the deep network $\mathcal{F}$ still needs to make predictions at all locations, these hard-to-fix errors caused by the local variations may further lead to uninformative gradient updates that increase errors at other easier-to-predict locations.

To address the problems introduced by irrelevant spectral variations, we propose to reduce their impact using an *"ignore-and-interpolate"* correction strategy. The idea is that, rather than having the variations pollute the results, we can let the network "ignore" predictions at the locations being affected by variations and re-fill their values by doing interpolation with nearby predictions within a spatially-adjacent neighborhood S.

Denote the deep network as $\mathcal{F}$, and $\mathcal{F}_{\mathcal{H}}$ and $\mathcal{F}_{\mathcal{C}}$ as two sub-branches (may have shared layers) that predict preliminary heights $\hat{\mathbf{Y}}_i$ and confidence scores $\hat{\mathbf{C}}_i$, respectively. The correction layer of the network updates height values at low-confidence locations using high-confidence values to produce the output heights $\hat{\mathbf{Y}}_i^*$. In the following, we explain the details of the key steps on confidence estimation and confidence-based pseudo-interpolation.

Confidence estimation: The goal of confidence estimation is to help the model locate predictions with potential high-errors. We estimate the confidence using maximum likelihood estimation (MLE). The normal-based MLE can be estimated by assuming each of the predictions (i.e., each pixel from an input image patch) follows a normal distribution, i.e, $p(\mathbf{x}_j) \sim \mathcal{N}(\hat{y}_j, (1/\hat{c})^2)$, where predicted heights $\hat{y}_j \in \hat{\mathbf{Y}}_i$ and the inverses of confidences $\hat{c}_j \in \hat{\mathbf{C}}_i$ are used as means and standard deviations, respectively; and $\hat{\mathbf{Y}}_i$ and $\hat{\mathbf{C}}_i$ are in $\mathbb{R}^{m\times m\times 1}$, which have the same shape as inputs $\mathbf{X}_i$ as specified in the problem definition (Sec. 2). Denote $\mathbf{\Theta}_{\mathcal{H}}$ and $\mathbf{\Theta}_{\mathcal{C}}$ as network parameters for $\mathcal{F}_{\mathcal{H}}$ and $\mathcal{F}_{\mathcal{C}}$, respectively, and so we have $\hat{\mathbf{Y}} = \mathcal{F}_{\mathcal{H}}(\mathbf{X}_i, \mathbf{\Theta}_{\mathcal{H}})$ and $\hat{\mathbf{C}} = \mathcal{F}_{\mathcal{C}}(\mathbf{X}_i, \mathbf{\Theta}_{\mathcal{C}})$. Using pixels from each input image patch $(\mathbf{x}_j, y_j) \in (\mathbf{X}_i, \mathbf{Y}_i)$ as examples, the optimization is then:

$$\arg\min_{\mathbf{\Theta}_{\mathcal{H}},\mathbf{\Theta}_{\mathcal{C}}} \mathcal{L}_{MLE}(\hat{\mathbf{Y}}_i|\mathbf{X}_i, \mathbf{\Theta}_{\mathcal{H}}, \mathbf{\Theta}_{\mathcal{C}}) = -\log \prod_{j=1}^{M} p(\hat{y}_j|\mathbf{x}_j, \mathbf{\Theta}_{\mathcal{H}}, \mathbf{\Theta}_{\mathcal{C}})$$

$$= \arg\min_{\mathbf{\Theta}_{\mathcal{H}},\mathbf{\Theta}_{\mathcal{C}}} \sum_{j=1}^{M} \left[\frac{(y_j - \mathcal{F}_{\mathcal{H}}(\mathbf{x}_j, \mathbf{\Theta}_{\mathcal{H}}))^2}{2 \cdot (1/\mathcal{F}_{\mathcal{C}}(\mathbf{x}_j, \mathbf{\Theta}_{\mathcal{C}}))^2} + \log(\frac{1}{\mathcal{F}_{\mathcal{C}}(\mathbf{x}_j, \mathbf{\Theta}_{\mathcal{C}})}) \right]$$

where $p(\hat{y}_j|\mathbf{x}_j) = \frac{1}{\sqrt{2\pi}/\mathcal{F}_{\mathcal{C}}(\mathbf{x}_j,\mathbf{\Theta}_{\mathcal{C}})} \cdot \exp(-\frac{(y_j - \mathcal{F}_{\mathcal{H}}(\mathbf{x}_j,\mathbf{\Theta}_{\mathcal{H}}))^2}{2\cdot(1/\mathcal{F}_{\mathcal{C}}(\mathbf{x}_j,\mathbf{\Theta}_{\mathcal{C}}))^2})$ and $M = m^2$ is the number of pixels per image patch. The first term shows that the errors on pixels with lower confidences (i.e., higher variances) will be scaled down, and the second term naturally constraints the variances.

Pseudo-Interpolation: We use a confidence-based interpolation layer, which is attached after the preliminary height predictions $\hat{\mathbf{Y}}_i$, to generate the output heights $\hat{\mathbf{Y}}_i^*$ using $\hat{\mathbf{C}}_i$. We call it pseudo-interpolation as the approach we use is different from traditional interpolation where known values are used to fill in unknown values based on explicit spatial distances. In our case, all values are known but with different

levels of confidences in $\hat{\mathbf{C}}_i$. Additionally, we do not use explicit distance-based interpolation (e.g., calculating distances to all points and then do a weighted average), as that process is often non-differentiable. Since the confidence-based pseudo-interpolation will be used as a **sub-routine** in the entire framework (Sec. 4.2), we need it to be differentiable to better co-learn confidences and heights for better final outputs. Specifically, our pseudo-interpolation uses local neighborhoods S_j around each pixel, where S_j is a $W \times W$ window. We use 7×7 window for S_j by default. The new values are weighted averages of all values in S_j:

$$\hat{\mathbf{Y}}_i^* = \big((\hat{\mathbf{Y}}_i \odot \hat{\mathbf{C}}_i) * \mathbf{J}_W\big) \oslash \big(\hat{\mathbf{C}}_i * \mathbf{J}_W\big) \tag{1}$$

where $\mathbf{J}_W$ is a $W \times W$ matrix with all ones; $\odot$ and $\oslash$ are Hadamard product and division, respectively; and $*$ is the convolutional operation with the latter as the kernel.

Here $\hat{\mathbf{Y}}_i^*$ is not the final output, and will be iteratively updated together with the confidence $\hat{\mathbf{C}}_i$ in the coming section.

4.2 Recurrent Interpolation for Self-Adaptive Refinement

As after each pseudo-interpolation the confidence values should be changed at each pixel, we propose a recurrent interpolation structure to allow the predictions to continue to refine themselves based on the new heights $\hat{\mathbf{Y}}_i^*$. Specifically, here we use the pseudo-interpolation as a sub-routine in a recurrent manner. After each round of executing $\mathcal{F}_\mathcal{H}$, we feed the resulting $\hat{\mathbf{Y}}_i^*$ – together with the original features $\mathbf{X}_i$ – as inputs back to $\mathcal{F}_\mathcal{H}$ and $\mathcal{F}_\mathcal{C}$ for the next round of refinement. In other words, here we include $\hat{\mathbf{Y}}_i^*$ as an additional input to $\mathcal{F}_\mathcal{H}$ and $\mathcal{F}_\mathcal{C}$ as compared to the base version in Sec. 4.1:

- **Initial inputs:** Denote $\hat{\mathbf{Y}}_i^{*,t}$ as the output after the t^{th} round. Given the new input structure, we need to provide an initial input for $\hat{\mathbf{Y}}_i^{*,0}$ for the very first round. Here we use the narrow-footprint height $\mathbf{Z}_i$ as $\hat{\mathbf{Y}}_i^{*,0}$, which is part of the known inputs to this problem (Sec. 2) and provides a peek of true heights in the patch.
- **Number of recurrent steps:** In test phase, the recurrent procedure can run till convergence, i.e., when:

  $$\mathbf{e}^T \cdot \big((\hat{\mathbf{Y}}_i^{*,t} - \hat{\mathbf{Y}}_i^{*,t-1}) \oslash \hat{\mathbf{Y}}_i^{*,t-1}\big) \cdot \mathbf{e} \leq \tau$$

  where $\mathbf{e}$ is a vector of ones, and τ is a user-set tolerance. It can be constrained by a maximum number of iterations (e.g., 5, 10). For training, we fix the number of recurrent steps to 5 to reduce overhead.
- **Training:** We evaluate the MLE loss $\mathcal{L}_{MLE}$ at each recurrent step and update the parameters after the final recurrent step using the average of the losses (Alg. 1).

4.3 Truth-Based Correction with Narrow-Footprint Heights

Although the narrow-footprint heights $\mathbf{Z}_i$ is used as an initial seed for height expansion in the recurrent framework, the training has not yet explicitly utilized its truth-nature to more effectively correct the predictions $\hat{\mathbf{Y}}_i^*$. This section aims to perform such explicit and truth-based correction with an inverse-confidence-based regression layer $\mathcal{F}_R$.

$\mathcal{F}_R$ replaces the last layer of the height-prediction branch $\mathcal{F}_\mathcal{H}$ to perform the correction. To avoid confusion, $\mathcal{F}_R$ becomes the new final layer of $\mathcal{F}_\mathcal{H}$, which happens before the pseudo-interpolation (lines 9-10 in Alg. 1) and is included as part of the recurrent process. $\mathcal{F}_R$ takes the features learned from its previous layer and performs a self-optimized linear combination using weights $\mathbf{\Theta}_R$ to generate the outputs $\hat{\mathbf{Y}}_i$. In other words, while $\mathcal{F}_R$ is part of the whole network, its weights $\mathbf{\Theta}_R$ are obtained by a direct least-squares type of solutions instead of by gradient descent, which has been used as a paradigm in [Bertinetto *et al.*, 2018] for adaptation. We use this paradigm rather than standard gradient descent because: (1) It provides a direct one-step correction to allow the predictions to quickly adapt to the limited ground truth $\mathbf{Z}_i$ in-network; (2) The least-squares solution can be computed with a closed-form solution during feed-forward, which does not interfere with or introduce major overhead to the overall gradient descent process of $\mathcal{F}_\mathcal{H}$ and $\mathcal{F}_\mathcal{C}$; and (3) Because it can be seamlessly injected into the network, the network can learn weights to adapt to additional correction from the regression layer $\mathcal{F}_R$.

Note that this linear optimization is independent for each input image patch $\mathbf{X}_i$, and as $\mathbf{\Theta}_R$ for $\mathcal{F}_R$ is not updated in back-propagation, we can also consider weights in $\mathbf{\Theta}_R$ as auxiliary inputs, which are dynamically updated on-the-fly. In addition, the samples only involve the pixels intersecting with the narrow-footprints from the scan lines (Fig. 1, stripes in $\mathbf{Z}_i$) where ground-truth heights are known as inputs. For simplicity, denote G as the set of pixels that have narrow-footprint ground-truth given, and $|G|$ as the set cardinality. Further, denote $\mathbf{H}_i(G) \in \mathbb{R}^{|G| \times (h+1)}$ as a list containing h learned features from the previous layer of $\mathcal{F}_R$ for all pixels in $|G|$; we add an additional "1" value to each row to learn the bias term. Similarly, denote $\hat{\mathbf{C}}_i(G)$ and $\mathbf{Z}_i(G)$ in $\mathbb{R}^{|G| \times 1}$ as the confidence and known-heights (inputs), respectively. The weights $\mathbf{\Theta}_R \in \mathbb{R}^{h+1}$ are then learned as:

$$\begin{aligned}
\mathbf{\Theta}_R =& \arg\min_{\mathbf{\Theta}_R} ||\big(\mathbf{Z}_i(G) - \mathbf{H}_i(G)\mathbf{\Theta}_R\big)^T \cdot \mathrm{diag}(\hat{\mathbf{C}}_i(G)^{-1}) \cdot \\
& \big(\mathbf{Z}_i(G) - \mathbf{H}_i(G)\mathbf{\Theta}_R\big) + \lambda \mathbf{\Theta}_R^T \mathbf{\Theta}_R||_2^2 \\
=& (\mathbf{H}_i(G)^T \mathrm{diag}(\hat{\mathbf{C}}_i(G)^{-1})\mathbf{H}_i(G) + \lambda I)^{-1} \mathbf{H}_i(G)^T \mathbf{Z}_i(G) \\
=& (\mathbf{R}_i^T \mathbf{Q}_i^T \mathrm{diag}(\hat{\mathbf{C}}_i(G)^{-1})\mathbf{Q}_i\mathbf{R}_i + \lambda I)^{-1} \mathbf{R}_i^T \mathbf{Q}_i^T \mathbf{Z}_i(G) \\
=& (\mathbf{R}_i^T \mathrm{diag}(\hat{\mathbf{C}}_i(G)^{-1})\mathbf{R}_i + \lambda I)^{-1} \mathbf{R}_i^T \mathbf{Q}_i^T \mathbf{Z}_i(G)
\end{aligned}$$

where λ is the scaling factor for ridge regression, and $\mathbf{H}_i = \mathbf{Q}_i\mathbf{R}_i$ is the QR factorization of $\mathbf{H}_i$ for numerical stability. We use inverse confidence to weigh pixels in the ridge regression to strengthen the correction on the potentially underperforming ones.

Finally, to make the correction more stable, we perform a random ensemble on the linear optimization step. Specifically, we split G into k overlapping subsets (e.g., 3 subsets where each contains 2/3 of points in G), and perform k sepa-

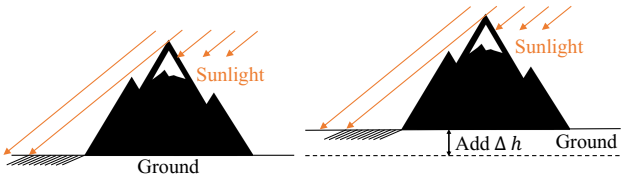


Figure 2: Similar spectral patterns (shadows) at various base heights.

rate linear optimizations of $\mathbf{\Theta}_R$. The ensemble is then:

$$\hat{\mathbf{Y}}_i^{ens} = \frac{1}{k} \cdot \sum_{j=1}^{k} \mathcal{F}_R(\mathbf{H}_i, \mathbf{\Theta}_R^j) \tag{2}$$

Alg. 1 illustrates the key steps in Sec. 4.1 to 4.3 using one batch as an example.

Algorithm 1 An iteration in spatial self-corrective learning

Require: A batch B of $\{(\mathbf{X}_i, \mathbf{Z}_i, \mathbf{Y}_i)\}$ with M pixels per image.
for $(\mathbf{X}_i, \mathbf{Z}_i, \mathbf{Y}_i)$ in B **do**
  $\hat{\mathbf{Y}}_i^{*,0}$ = Init($\mathbf{Z}_i$)
  {Recurrent correction. Network notation for convenience: $\mathcal{F}_\mathcal{H} = \mathcal{F}'_\mathcal{H} + \mathcal{F}_R$ (last correction layer), with $\mathbf{\Theta}_\mathcal{H} = \mathbf{\Theta}'_\mathcal{H} \cup \mathbf{\Theta}_R$; and $\mathcal{F}_\mathcal{C}$ has parameters $\mathbf{\Theta}_\mathcal{C}$.}
  for t = 1 to T **do**
    $\mathbf{H}_i^t = \mathcal{F}'_\mathcal{H}(\mathbf{X}_i, \hat{\mathbf{Y}}_i^{*,t-1}, \mathbf{\Theta}'_\mathcal{H})$
    $\hat{\mathbf{C}}_i^t = \mathcal{F}_\mathcal{C}(\mathbf{X}_i, \hat{\mathbf{Y}}_i^{*,t-1}, \mathbf{\Theta}_\mathcal{C})$
    for j = 1 to k **do**
      $\mathbf{\Theta}_R^j$ = Linear_Opt($\mathbf{H}_i^t$, $\hat{\mathbf{C}}_i^t$, $\mathbf{Z}_i$, G) {G are pixels intersecting scan lines}
    end for
    $\hat{\mathbf{Y}}_i^{ens,t} = \frac{1}{k} \cdot \sum_{j=1}^{k} \mathcal{F}_R(\mathbf{H}_i^t, \mathbf{\Theta}_R^j)$
    $\hat{\mathbf{Y}}_i^{*,t}$ = Conf_Pseudo_Interpolate($\hat{\mathbf{Y}}_i^{ens,t}$, $\hat{\mathbf{C}}_i^t$)
  end for
  $\mathcal{L}_{MLE}^i = \sum_{t=1}^{T} \mathcal{L}_{MLE}(\mathbf{Y}_i, \hat{\mathbf{Y}}_i^{*,t}, \hat{\mathbf{C}}_i^t)/(T \cdot M)$
end for
Update $\mathbf{\Theta}'_\mathcal{H}$ and $\mathbf{\Theta}_\mathcal{C}$ using $\sum_{i=1}^{|B|} \mathcal{L}_{MLE}^i / |B|$

4.4 Height Augmentation

While the spectral variations in each scene may reflect height variations, they tend to be limited in estimating the base height of the scene. More specifically, we can decompose the heights of a scene into two parts $\mathbf{Y}_i = \mathbf{Y}_i^{var} + \beta$, where β is a constant representing a base height. Imagine if we increase or decrease β by 100m (Fig. 2, we may obtain very similar spectral variations (e.g., shadows) as the lights from the sun can be considered very much as parallel beams with similar angles at different altitudes.

While the limited observations in $\mathbf{Z}_i$ help, the network may still overfit to certain observed height values. Thus, we include a simple base-height augmentation, where we randomly increase or decrease the ground-truth height (identical for $\mathbf{Y}_i$ and $\mathbf{Z}_i$) of different samples during training. The random change range is bounded by 500m as over-large altitude changes may no longer keep similar characteristics in some spectra. The augmentation encourages the network to focus more on predicting variations, which can then be combined with information in $\mathbf{Z}_i$ to generate the height predictions.

5 Experiments

5.1 Datasets

Height data: As the goal is to fill height data gaps and generate large-footprint height data, ideally we also need such data for evaluation. Due to the expensive collection cost, large-footprint height information is in general less available at large scales. Among existing datasets, the most commonly used height data are provided by the Shuttle Radar Topography Mission (SRTM). Thus, we use the SRTM data as a main source for our experiments. As SRTM mainly covers regions between 60°N and 56°S latitudes, we additionally include the Interferometric Synthetic Aperture Radar (IFSAR) dataset, which offers elevation measurements in the Arctic regions in Alaska. Finally, we include test data from the most recent ICESat-2 satellites. Since ICESat-2 only provides narrow-footprint height observations (e.g., Fig. 1), in the testing we leave out data from a subset of the scan lines (i.e., 2 out of 6) per image patch for evaluation.

Spectral imagery: For the spectral imagery data, we need to use images that are temporally aligned with the data from SRTM, IFSAR and ICESat-2. Among them, SRTM and IFSAR were collected in 2000 and 2010, respectively, and the ICESat-2 mission started in 2018. Given the temporal distribution, we use multispectral imagery from Landsat-7, which covers all the durations. Especially, for SRTM and IFSAR, Landsat-7 was one of the few high-resolution multispectral satellite missions that overlapped with their timestamps (e.g., other popular missions such as Landsat-8 and Sentinel-2 started in 2013 and 2015, respectively).

Additional data details: As most of the bands in Landsat-7 imagery have a spatial resolution at 30m (side-length of a pixel), we use 30m as the resolution in all experiments. All height data have equal or higher resolution, so they can be safely resampled to 30m. We select 5 locations in high-latitude regions with different landscapes: (1) Two locations for SRTM in the northwestern US (i.e., Washington and Idaho in Table 1), which are mountainous regions with larger height variations (e.g., steep mountainsides, deep valleys) and more forests. Each region is about 22,500 km^2 in size. (2) Two locations for ISFAR in the Arctic region within northern Alaska, US (Alaska-1 and Alaska-2 in Table 1). One location has a bare-earth type of landscape with limited snow cover, and the other is mostly covered by snow. Each region is about 3,600 km^2 in size as ISFAR has smaller footprints. (3) Two locations for ICESat-2 in Alaska, US (Alaska-3) and Yukon, Canada, with bare-earth landscapes. Each location is about 2,000 km^2 in size, which follows along a segment of the trajectory of an ICESat-2 satellite. Finally, all the data are split and formatted into image patches of size 500×500 (Sec. 2).

Training and testing: We group SRTM and ISFAR together for the first evaluation as we have ground-truth large-footprint height data. We use four train-test splits, each with three locations as training and one as testing. Numbers in Tables 1 and 2 represent results with the location in the corresponding column being the test site. To simulate narrow-footprint height observations at intersecting scan lines (inputs $\mathbf{Z}_i$, Fig. 1), we sample 2 sets of 3-lines (ICESat-2 scans three

	Base	DA	FNN	Kriging	Ridge	RF	KCN	KCN-att	KCN-sage	SCL
Washington	0.174	0.184	0.253	0.217	0.471	0.543	0.252	0.260	0.286	**0.131**
Idaho	0.065	0.062	0.084	0.061	0.517	0.493	0.089	0.080	0.077	**0.033**
Alaska-1	0.115	0.157	0.158	0.167	0.245	0.413	0.184	0.191	0.204	**0.075**
Alaska-2	0.225	0.972	0.278	0.284	0.791	0.923	0.304	0.318	0.259	**0.216**
Alaska-3	0.213	0.183	0.203	0.242	0.353	0.270	0.391	0.490	0.947	**0.135**
Yukon	0.190	0.244	0.183	0.184	0.179	0.180	0.187	0.180	0.228	**0.145**
Mean	0.164	0.301	0.193	0.193	0.426	0.470	0.235	0.253	0.334	**0.123**

Table 1: sMAPEs for height interpolation.

	Base	DA	FNN	Kriging	Ridge	RF	KCN	KCN-att	KCN-sage	SCL
Washington	0.806	0.795	0.584	0.631	0.565	0.675	0.494	0.487	0.322	**0.843**
Idaho	0.772	0.787	0.634	0.810	0.772	0.718	0.764	0.776	0.678	**0.974**
Alaska-1	0.900	0.809	0.793	0.763	0.673	0.346	0.692	0.665	0.568	**0.95**
Alaska-2	0.744	0.025	0.694	0.588	0.521	0.250	0.566	0.534	0.580	**0.905**
Alaska-3	0.879	0.894	0.769	0.745	0.839	0.883	0.478	0.042	0.487	**0.901**
Yukon	0.142	0.130	0.292	0.397	0.318	0.224	0.314	0.442	-0.018	**0.459**
Mean	0.707	0.573	0.628	0.656	0.615	0.516	0.551	0.491	0.436	**0.839**

Table 2: Correlation coefficients for height interpolation.

lines simultaneously) in two different directions with equal intervals (3km; same as ICESat-2). During testing, all models are first fine-tuned with $\mathbf{Z}_i$ before prediction. For ICESat-2, since there is no large-footprint height data, we use a trained model from ISFAR(Alaska-1) and fine-tune it with 4 out of 6 lines per scene and use the rest for testing.

5.2 Methods for Evaluation

The following baseline methods are used in the comparison:

- **Base:** The base network is a U-Net architecture with 3 encoding blocks and 3 decoding blocks to learn features across multiple scales and gradually combines them to achieve full-resolution height prediction.
- **DA:** The base network with domain adaptation by using a generative adversarial network for adversarial learning [Goodfellow *et al.*, 2020; Tzeng *et al.*, 2017]. DA learns domain-invariant features, but is limited when landscapes are very different in training and testing [Li *et al.*, 2023].
- **FNN:** A 5-layer fully-connected network that estimates height values based on spectral features at the pixel-level.
- **Kriging:** The universal Kriging interpolation [Cressie, 2015] that uses spectral features and heights on the narrow-footprint surface heights along scan lines (Fig. 1(a)) as inputs to predict the unknown height at other locations. We use the closest 50 points as the neighborhood.
- **Ridge:** The ridge regression using L2-regularization with the scaling factor λ of 0.1. It is limited for non-linear and noisy problems [Bao *et al.*, 2022].
- **RF:** Random forest regression with 100 trees using spectral features at pixel-levels.
- **KCN:** The Kriging convolutional network [Appleby *et al.*, 2020] that interpolates height values based on known values on narrow-footprint surface heights along scan lines (Fig. 1(a)) for each image. It relies on Graph Convolutional Network (GCN) to embed neighborhood information. We use the code provided by the authors in our experiments [KCN, 2019]. We keep the recommended hyperparameters and set the number of neighbors to use for interpolation to 50 (default was 5, which led to lower-quality results).
- **KCN-att:** KCN that uses graph attention [Veličković *et al.*, 2017] instead of original GCN layers to compute the attention weights of a node's neighbors.
- **KCN-sage:** KCN that uses GraphSAGE [Hamilton *et al.*, 2017] to learn aggregator functions of nodes.
- **SCL:** Our proposed spatial self-corrective learning method.

Metrics. We use the symmetric mean absolute percentage error (sMAPE) and correlation coefficients to evaluate the performances of the methods. We use sMAPE instead of MAPE as there are many lower-valued heights (e.g., near 0), which causes instability in MAPE calculation [Xie *et al.*, 2023a]. sMAPE is a standard extension of MAPE, which includes predicted values in the denominators to constrain the range to [0,1].

5.3 Results

Comparative Analysis

The neural-network-based models are trained with the Adam optimizer with an initial learning rate of 10^{-4}. From Tables 1 and 2, we can see that the proposed spatial SCL approach outperformed the baseline methods for all the scenarios in the experiments. Domain adaptation methods did not improve upon base networks mainly because different landscapes may have different height variation patterns [Li *et al.*, 2023]. Therefore, forcing a similar representation may hurt the ability to adapt to various landscapes. Pixel-wise regressors, such as FNN, ridge regression, and random forest perform poorly, as they can be largely affected by the spectral variations that are irrelevant to height changes. They also have limited ability to adapt to new regions. Kriging performs better than the

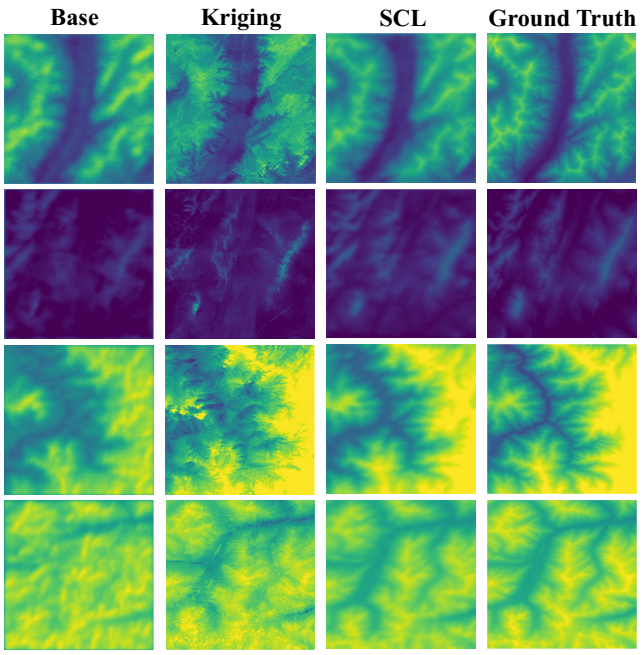


Figure 3: Example visualizations of height estimations.

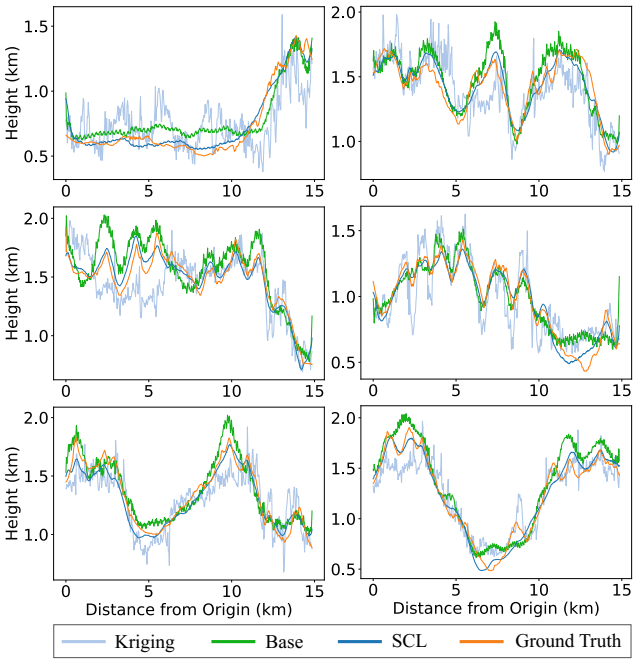


Figure 4: Height profile comparison.

	Base	No-RR	No-Reg	No-Rec	SCL
sMAPE	0.174	0.488	0.210	0.165	**0.131**
Corr.	0.806	0.503	0.773	0.831	**0.843**

Table 3: Effects of different components in SCL.

pixel-wise regressors in most cases as it exploits the known height values in its neighborhood, which provide necessary guidance for the unlabeled locations. However, Kriging convolutional networks did not show much improvements over, and sometimes performed worse than, Kriging in this problem, as the major limiting factor of both types of methods may be the spatially-constrained distribution of known values, under which the enhancement from the additional nonlinear capacity in KCN's feature learning becomes limited.

Comparing across regions, most methods perform the best in Idaho, which has relatively sharper landscapes and visual features that make the estimation easier. In contrast, Alaska-2 and Yukon were more challenging for all methods. This is potentially caused by the more flat landscapes with smaller height variations, which may reduce the amount of useful features (e.g., shadows).

Fig. 3 visualizes several examples of height estimation in several areas. We can see SCL can better capture the high mountains and deeper valleys, and at the same time keep the terrains smoother (i.e., less affected by irrelevant height variations) with the confidence-based interpolation. Fig. 4 shows detailed height profiles along randomly sampled straight lines intersecting the regions. Similarly, we can see that the predictions from SCL show the best alignment with the ground truth compared to the others.

Ablation Studies

We conducted ablation experiments to validate the effectiveness of the proposed recurrent refinement and regression-based correction. In the first scenario, we removed both recurrent refinement and regression-based correction (No-RR) from the proposed model. The models were tested on the Seattle region and trained on the other three regions. In the second scenario, we replaced the inverse-confidence-based regression layer with a trainable 1×1 convolutional layer (No-Reg) for height regression. In the third scenario, we set the number of recurrent steps to 1 (No-Rec) to remove the effect of recurrent refinement while keeping other settings constant. The results in Table 3 show that the removal of either recurrent refinement or regression-based correction leads to a drop in performance.

6 Conclusions

We proposed a spatial self-corrective learning framework to expand the coverage of surface height data in high latitudes by combining narrow-footprint heights and large-footprint satellite imagery. The new framework includes several key components, i.e., confidence-based pseudo-interpolation, recurrent self-correction, and truth-based correction with a regression layer. The proposed method demonstrated consistent improvements over baseline methods in the experiments for different landscapes.

Our future work will evaluate the approach for more types of landscapes and explore the integration of the methods in domain analysis pipelines. We will also develop a benchmark dataset to facilitate future comparisons on this problem. Finally, this work has not considered the issues of cloud coverage in satellite imagery, which may need further exploration especially for the polar regions [Xie *et al.*, 2023b].

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 2105133, 2126474 and 2147195; NASA under Grant No. 80NSSC22K1164 and 80NSSC21K0314; USGS under Grant No. G21AC10207; Google's AI for Social Good Impact Scholars program; the DRI award and the Zaratan supercomputing cluster at the University of Maryland; and Pitt Momentum Funds award and CRC at the University of Pittsburgh. We also thank Dr. Sinead Farrell for the inputs on ICESat-2 data.

References

[Amirkolaee and Arefi, 2019] Hamed Amini Amirkolaee and Hossein Arefi. Height estimation from single aerial images using a deep convolutional encoder-decoder network. *ISPRS journal of photogrammetry and remote sensing*, 149:50–66, 2019.

[Appleby *et al.*, 2020] Gabriel Appleby, Linfeng Liu, and Li-Ping Liu. Kriging convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3187–3194, 2020.

[Bao *et al.*, 2022] Han Bao, Xun Zhou, Yiqun Xie, Yingxue Zhang, and Yanhua Li. Covid-gan+: Estimating human mobility responses to covid-19 through spatio-temporal generative adversarial networks with enhanced features. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(2):1–23, 2022.

[Bertinetto *et al.*, 2018] Luca Bertinetto, Joao F Henriques, Philip HS Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. *arXiv preprint arXiv:1805.08136*, 2018.

[Caballero *et al.*, 2013] William Caballero, Ramón Giraldo, and Jorge Mateu. A universal kriging approach for spatial functional data. *Stochastic environmental research and risk assessment*, 27:1553–1563, 2013.

[Chen *et al.*, 2018] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[Cressie, 2015] Noel Cressie. *Statistics for spatial data*. John Wiley & Sons, 2015.

[Farr and Kobrick, 2000] Tom G Farr and Mike Kobrick. Shuttle radar topography mission produces a wealth of data. *Eos, Transactions American Geophysical Union*, 81(48):583–585, 2000.

[Goodfellow *et al.*, 2020] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[Hamilton *et al.*, 2017] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

[KCN, 2019] KCN. The implementation of Kriging Convolutional Networks algorithm. https://github.com/tufts-ml/KCN, 2019. Accessed: 2/26/2023.

[Li *et al.*, 2023] Zhili Li, Yiqun Xie, Xiaowei Jia, Kara Stuart, Caroline Delaire, and Sergii Skakun. Point-to-region co-learning for poverty mapping at high resolution using satellite imagery. In *AAAI Conference on Artificial Intelligence*, 2023.

[Liu *et al.*, 2017] Chun Liu, Xin Huang, Dawei Wen, Huijun Chen, and Jianya Gong. Assessing the quality of building height extraction from ziyuan-3 multi-view imagery. *Remote Sensing Letters*, 8(9):907–916, 2017.

[Liu *et al.*, 2022] Xiaoqiang Liu, Yanjun Su, Tianyu Hu, Qiuli Yang, Bingbing Liu, Yufei Deng, Hao Tang, Zhiyao Tang, Jingyun Fang, and Qinghua Guo. Neural network guided interpolation for mapping canopy height of china's forests by integrating gedi and icesat-2 data. *Remote Sensing of Environment*, 269:112844, 2022.

[McKinley and Levine, 1998] Sky McKinley and Megan Levine. Cubic spline interpolation. *College of the Redwoods*, 45(1):1049–1060, 1998.

[Pörtner *et al.*, 2019] Hans-Otto Pörtner, Debra C Roberts, Valérie Masson-Delmotte, Panmao Zhai, Melinda Tignor, Elvira Poloczanska, and NM Weyer. The ocean and cryosphere in a changing climate. *IPCC special report on the ocean and cryosphere in a changing climate*, 1155, 2019.

[Qin, 2019] Rongjun Qin. A critical analysis of satellite stereo pairs for digital surface model generation and a matching quality prediction model. *ISPRS Journal of Photogrammetry and Remote Sensing*, 154:139–150, 2019.

[Richter-Menge *et al.*, 2019] J Richter-Menge, ML Druckenmiller, MO Jeffries, et al. Arctic ecosystems and communities are increasingly at risk due to continued warming and declining sea ice. 2019.

[Ronneberger *et al.*, 2015] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[Shepard, 1968] Donald Shepard. A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM national conference*, pages 517–524, 1968.

[Soergel *et al.*, 2009] Uwe Soergel, Eckart Michaelsen, Antje Thiele, Erich Cadario, and Ulrich Thoennessen. Stereo analysis of high-resolution SAR images for building height estimation in cases of orthogonal aspect directions. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64(5):490–500, 2009.

[Tzeng *et al.*, 2017] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative

domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017.

[UN, 2022] UN. United Nations' sustainable development goals. https://sdgs.un.org/goals, 2022. Accessed: 11/30/2022.

[Veličković *et al.*, 2017] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[Wackernagel and Wackernagel, 2003] Hans Wackernagel and Hans Wackernagel. Ordinary kriging. *Multivariate Geostatistics: An Introduction with Applications*, pages 79–88, 2003.

[Wu *et al.*, 2021] Yuankai Wu, Dingyi Zhuang, Aurelie Labbe, and Lijun Sun. Inductive graph neural networks for spatiotemporal kriging. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 4478–4485, 2021.

[Xie *et al.*, 2021] Yiqun Xie, Erhu He, Xiaowei Jia, Han Bao, Xun Zhou, Rahul Ghosh, and Praveen Ravirathinam. A statistically-guided deep network transformation and moderation framework for data with spatial heterogeneity. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 767–776. IEEE, 2021.

[Xie *et al.*, 2023a] Yiqun Xie, Weiye Chen, Erhu He, Xiaowei Jia, Han Bao, Xun Zhou, Rahul Ghosh, and Praveen Ravirathinam. Harnessing heterogeneity in space with statistically guided meta-learning. *Knowledge and information systems*, 65(6):2699–2729, 2023.

[Xie *et al.*, 2023b] Yiqun Xie, Zhili Li, Han Bao, Xiaowei Jia, Dongkuan Xu, Xun Zhou, and Sergii Skakun. AutoCM: Unsupervised deep learning for satellite imagery composition and cloud masking using spatio-temporal dynamics. In *AAAI Conference on Artificial Intelligence*, 2023.