A Co-Training Approach for Spatial Data Disaggregation

João Monteiro
Universidade de Lisboa, IST/INESC-ID
Lisboa, Portugal
joao.miguel.monteiro@tecnico.ulisboa.pt

Bruno Martins
Universidade de Lisboa, IST/INESC-ID
Lisboa, Portugal
bruno.g.martins@tecnico.ulisboa.pt

Miguel Costa
Vodafone
Lisboa, Portugal
miguel.costa2@vodafone.com

João M. Pires
Universidade NOVA de Lisboa, FCT/NOVA-LINCS
Caparica, Portugal
jmp@fct.unl.pt

ABSTRACT

Socio-demographic information is usually only accessible at relatively coarse spatial resolutions. However, its availability at thinner granularities is of substantial interest for several stakeholders, since it enhances the formulation of informed hypotheses on the distribution of population indicators. Spatial disaggregation methods aim to compute these fine-grained estimates, often using regression algorithms that employ ancillary data to re-distribute the aggregated information. However, since disaggregation tasks are ill-posed, and given that examples of disaggregated data at the target geospatial resolution are seldom available, model training is particularly challenging. We propose to address this problem through a self-supervision framework that iteratively refines initial estimates from seminal disaggregation heuristics. Specifically, we propose to co-train two different models, using the results from one model to train/refine the other. By doing so, we are able to explore complementary views from the data. We assessed the use of co-training with a fast regressor based on random forests that takes individual raster cells as input, together with a more expressive model, based on a fully-convolutional neural network, that takes raster patches as input. We also compared co-training against the use of self-training with a single model. In experiments involving the disaggregation of a socio-demographic variable collected for Continental Portugal, the results show that our co-training approach outperforms alternative disaggregation approaches, including methods based on self-training or co-training with two similar fully-convolutional models. Co-training is effective at exploring the characteristics of both regression algorithms, leading to a consistent improvement in different types of error metrics.

CCS CONCEPTS

• **Information systems** → Geographic information systems; • **Computing methodologies** → Semi-supervised learning settings.

KEYWORDS

geospatial data disaggregation, dasymetric disaggregation, self-supervised learning, co-training, encoder-decoder neural networks, convolutional neural networks, deep learning

ACM Reference Format:
João Monteiro, Bruno Martins, Miguel Costa, and João M. Pires. 2022. A Co-Training Approach for Spatial Data Disaggregation. In *The 30th International Conference on Advances in Geographic Information Systems (SIGSPATIAL '22), November 1–4, 2022, Seattle, WA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3557915.3561475

SIGSPATIAL '22, November 1–4, 2022, Seattle, WA, USA

ACM ISBN 978-1-4503-9529-8/22/11…$15.00
https://doi.org/10.1145/3557915.3561475

1 INTRODUCTION

Demographic and socio-economic statistics are usually only available or released at relatively aggregated levels corresponding to coarse and irregular geospatial regions (e.g., districts or municipalities). This is not ideal for analyzing the data through different partitions of space (e.g., through high-resolution regular tessellations of the geographic space, such as those present in the case of raster representations based on gridded cells), or in terms of their relation to particular terrain characteristics. In face of these restrictions, spatial disaggregation techniques can be used to provide more localized information, generating high-resolution estimates from count data made available at coarse geospatial resolutions.

Seminal disaggregation algorithms use straightforward strategies to translate the known counts associated to source administrative regions to raster cells at a given target resolution. For instance, mass-preserving areal weighting assumes that the aggregated values can be divided uniformly across the source regions [16], while pycnophylactic interpolation includes a degree of spatial autocorrelation in the variable being disaggregated [30]. More recent methods attempt to re-distribute the source counts with basis on regression analysis [8, 15, 21, 26] to weight the contribution of ancillary variables, e.g. using information on aspects such as land coverage, the location of buildings, or night-time light emissions. Although most previous literature used relatively simple regression algorithms (e.g., linear models), some authors have recently suggested that more advanced learning approaches (e.g., models based on CNNs) can improve the disaggregation performance [19, 22, 29].

Classical regression algorithms (e.g., linear models, or alternatives based on ensembles of decision trees) have been widely used for spatial data disaggregation. Approaches based on random forests are particularly suitable for dealing with some common characteristics of the data that are to be disaggregated, such as skewed distributions in the values or the presence of outliers [3]. Although these

methods can take direct advantage of ancillary features, they are limited in the sense that they process each raster cell independently. On the other hand, CNNs proposed for similar geospatial tasks (e.g., the encoder-decoder U-Net model, commonly used for tasks such as remote sensing image segmentation) can aid in exploring the intrinsic characteristics of geospatial data, namely the relationships between neighboring raster cells.

The training of spatial disaggregation methods is, nonetheless, an issue that requires particular attention. In fact, ground-truth data is rarely available to be directly used at the target resolution, e.g. for supporting model training and/or for evaluating model predictions. To overcome this, we propose to use a self-supervised learning strategy that iteratively refines the results from an initial disaggregation heuristic, by using the results of each iteration to inform the training of the following. Previous literature has used similar approaches, for instance by experimenting with self-training a single regression model [22]. In this article, we use co-training as a unified framework for taking advantage of the strengths associated with different regression algorithms. We specifically combine the advantages associated with fast cell-wise regressors, such as random forest models, together with more expressive alternatives that take into account the influence of neighboring cells, such as encoder-decoder CNNs for processing patches of raster data.

Our method was evaluated on a dataset collected for Continental Portugal, corresponding to the total amount, in thousands of euros, for the withdrawals on automated teller machines. We specifically experimented with the disaggregation of the information originally available at the level of coarse administrative regions, such as NUTS III units, into high-resolution grids with a resolution of 200 meters. We compared the proposed co-training approach, that alternates between two different regression algorithms, against alternative methods. These include seminal disaggregation heuristics, such as pycnophylactic interpolation, as well as the application of self-training using a single model. Previous work has specifically shown that the best disaggregation results for socio-demographic indicators can be obtained using different models, depending on the error metric being considered [22]. For instance, models based on CNNs are better in metrics such as the mean absolute error (MAE), while random forest alternatives are particularly competitive when analyzing results using metrics that over-penalize outliers, such as the root mean squared error (RMSE). By combining the two algorithms, we managed to consistently obtain the best values in all the considered error metrics, also improving upon the results obtained when using self-training with the corresponding individual models. For instance, when comparing co-training with the random forest model and the CNN, against self-training with the corresponding models, gains over the best baseline can increase from 15.6% to 18.3%, in MAE, and from 10.8% to 14.0%, in RMSE. Additionally, the training time required for the co-training approach is around 20% of the time for the alternative using self-training with the CNN.

In brief, the main contributions of this article are the following:

- We propose a co-training approach for spatial data disaggregation, alternating between two different regression algorithms to efficiently combine the sources of ancillary data;
- We compare the application of the co-training approach, using a random forest model and an encoder-decoder CNN, against the use of a self-training technique which only uses the corresponding single models. We also assessed the use of co-training with two similar models, namely two CNNs differing only in their initializations;
- We evaluate the co-training approach in the disaggregation of socio-demographic data concerning the territory of Continental Portugal. The experimental results show that co-training with two different models outperformed all the alternatives. Also, co-training with two versions of the same model leads to better results than self-training with the corresponding single model;
- We analyze model robustness to the presence of outliers in the data, as well as the error evolution across the iterations of the different self-supervised approaches.

The rest of this article is organized as follows: Section 2 presents the necessary background on spatial data disaggregation, including seminal methods and relevant related work. Section 3 describes our disaggregation approach. Section 4 describes the experimental setup, and Section 5 presents the results obtained for socio-demographic data relative to the territory of Continental Portugal. Finally, Section 6 highlights our main conclusions, as well as possible directions for future work.

2 BACKGROUND

This section starts by describing classical approaches for spatial data disaggregation. Then, it describes recent developments and practical applications, including methods which use regression analysis and/or some form of self-supervision.

2.1 Seminal Disaggregation Heuristics

Mass-preserving areal weighting is perhaps the simplest spatial disaggregation method. It divides the known counts associated to source administrative regions (e.g., the total values associated with coarse administrative districts) uniformly across their area [16]. Although this method is very straightforward, inherently conserving the total count values within each source region, it is based on the assumption that the phenomenon of interest is evenly distributed across the territory. Since most geospatial data are rarely uniform, it produces poor estimates when compared against the results of other alternative approaches.

Pycnophylactic interpolation is a refinement that starts by applying the mass-preserving areal weighting procedure, and then smooths the values for the resulting grid cells by replacing them with the average of their neighbors (i.e., the adjacent cells in a raster grid) [30]. The method continues until there are no significant changes from the previous iteration. The sum of all values within each source region is also kept consistent, in order to meet the mass-preserving property. The estimates obtained through pycnophylactic interpolation take into account the property of spatial auto-correlation, which states that regions close to each other tend to have similar values. However, the method does not enforce other properties about the distribution of the target variable, and often leads to over-smooth results.

If additional information is available on how the source data is geographically distributed, e.g. in the form of external variables expected to be correlated with the target counts, one can also use it

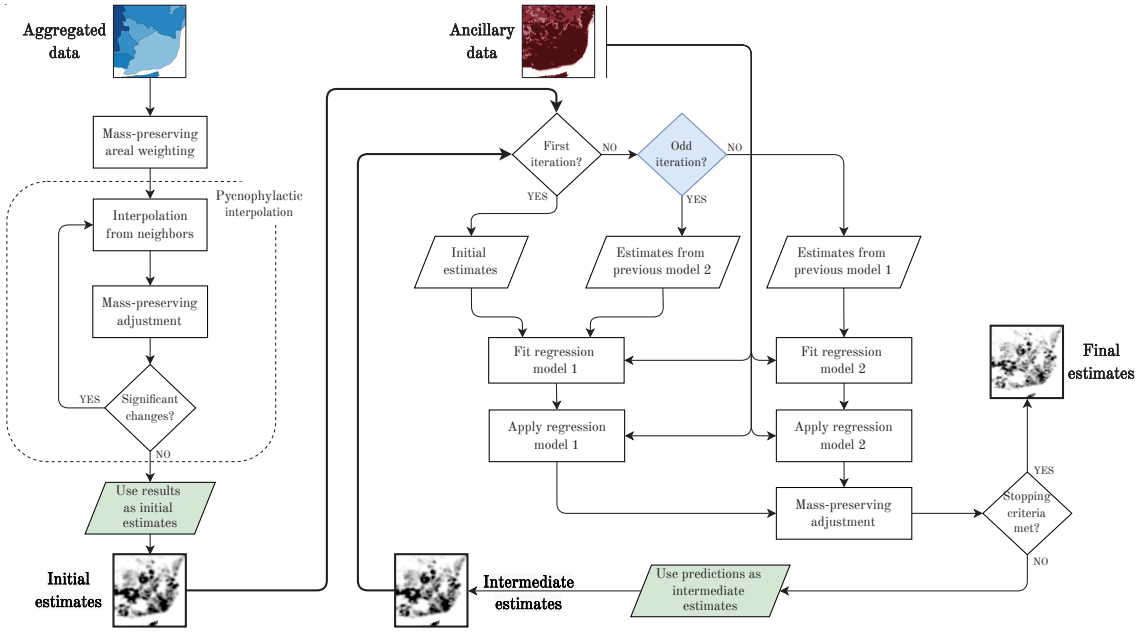

Figure 1: The different steps involved in the proposed co-training approach.

to inform the disaggregation. One can, for instance, use dasymetric mapping to distribute the source counts according to pre-specified weights, leveraging a source of ancillary data such as the presence of particular land cover classes (e.g., target counts should usually not be assigned to regions covered by water), or the human population distribution (i.e., most socio-economic variables correlate strongly with population density), to define the weights. In previous work, we have also combined dasymetric disaggregation, based on population density, with a smoothing process similar to that of pycnophylactic interpolation, this way promoting auto-correlation over the results, and calling it smooth weighted interpolation [22].

2.2 Spatial Disaggregation Using Regression

There are nowadays many openly available gridded datasets that consistently describe the human population distribution. These were often created using spatial disaggregation techniques that leverage machine learning procedures for combining ancillary information from different sources, within dasymetric approaches. Well-known examples include the datasets made available in the context of the WorldPop project. One study from Stevens et al. [28], within the scope of this project, reported the use of statistical modeling based on random forests to create gridded predictions of population density with a resolution of 100 meters. The authors used ancillary datasets that encode information on land coverage, digital elevation, the road network, and water bodies. First, their approach iteratively tunes a random forest model, at the level of census units, by predicting the population density with basis on mean values of the aggregated data. At the end of each iteration, importance values are computed for each feature, by measuring the change in model performance when ignoring information about that feature. The features with an importance equal to zero are removed, and the tuning process continues until only positive importance scores remain. The random forest model obtained with the selected features is then used for producing country-wide estimates at the cell level, leveraging the ancillary data available at the same resolution. Stevens et al. evaluated their method with census data for the countries of Cambodia, Vietnam, and Kenya. The predicted per-cell population densities were used to redistribute the data available at the level of census units, to obtain the per-cell counts. Then, the authors summed the values within small geospatial regions (e.g., villages or sub-locations), and compared the results with the corresponding known counts through metrics such as the Mean Absolute Error (MAE) or the Root Mean Squared Error (RMSE). The authors concluded that their method outperformed several competitors, such as the products produced within the Gridded Population of the World (GPW) [11] project.

Cheng et al. [9] reported another study that explores the use of dasymetric mapping based on regression analysis. The dasymetric weights were inferred by combining environmental information and mobile phone positioning data as the ancillary variables. Then, the weights were used for the disaggregation of census data for the territory of China, into a raster grid with a resolution of 1 kilometer, for each month in 2015. The method from Cheng et al. combines random forests with area-to-point kriging. The random forest model is trained with data at the town level, by predicting population density as the target variable, with basis on aggregated ancillary data (i.e., taking the mean values of each ancillary source, per town, as the independent variables). The learned model is then used to predict population estimates at the level of the target cells. The area-to-point kriging approach later adjusts the random forest predictions, leveraging the residuals computed for each town (i.e., computed by summing the encompassing residuals at the pixel

level). The pixel residuals are, in turn, computed through a weighted linear combination of the known residuals of neighboring towns (e.g., each pixel residual may increase based on the proximity to towns with higher residuals). The proposed approach achieved the best results in terms of the R^2 between predicted and real data, when comparing against gridded population products such as those from the WorldPop or GPW projects.

2.3 Self-Supervised Approaches

The training of the regression models that inform a disaggregation procedure constitutes a particular challenge. This is due to the lack of ideal ground-truth data, i.e., data available at the target resolution for at least a part of the territory. Some studies have proposed to address this problem through self-supervised approaches that operate directly at the target resolution, by iteratively refining results through the successive training of regression models.

Malone et al. [21] proposed dissever, an approach for downscaling soil organic carbon data using a regression algorithm based on generalized additive modeling. This approach fits, directly at the target resolution, a non-linear model associating the target variable and the predictive covariates. Dissever is initialized with a re-sampling procedure that transfers the data from the source to the target regions. This is followed by the iterative application of a generalized additive model, to predict the initial estimates from the set of covariates. In the iterative phase, results are first aggregated to the level of source zones (i.e., by averaging all the encompassing estimates), compared with the available values, and adjusted to keep consistency. Then, the generalized additive model is used to predict new values for all the grid cells.

Recent work from our team has adapted the dissever procedure, considering spatial disaggregation instead of the downscaling of non-additive variables. Monteiro et al. reported three studies that share a common general disaggregation methodology for combining pycnophylactic interpolation with dasymetric mapping [22–24]. In the first, Portuguese socio-economic variables were disaggregated with methods such as linear or generalized additive regression to combine the different sources of ancillary information. The second study instead focused on historical census data for the territories of the Netherlands, Belgium, and Great Britain, and leveraged more expressive regression algorithms, such as ensembles of decision trees and a neural network based on the LeNet-5 architecture [20]. The last study used a fully-convolutional neural network for the spatial disaggregation of socio-demographic data concerning the territory of Continental Portugal. The good results reported in the studies from Monteiro et al., and specifically the good performance of different regression models depending on the error metrics being considered, motivated the co-training approach reported in the present article. We also use the same CNN architecture outlined in the most recent study [22].

3 THE PROPOSED APPROACH

Our approach builds on previous methods for spatial data disaggregation based on the self-training of regression models, advancing over previous work by using co-training as a better alternative to model fitting. One can co-train two different regression models, this way assuring that different characteristics in the training data are explored. The original proposal for co-training from Blum and Mitchell [6] stated that, to guarantee success, the models being co-trained should explore two sufficient and redundant representations of the data that are conditionally independent given the target labels. Still, later studies have relaxed this assumption. For instance, Abney [1] showed that a weak dependence between two different views can also guarantee successful co-training, while Balcan et al. [2] discussed that the learner in each view should not be *confident but wrong*. Wang and Zhou [31], in turn, showed that if the two models have a large diversity (i.e., the diversity between the two learners is larger than their errors), co-training can also succeed. Some previous studies exploring co-training have successfully used two different supervised learning algorithms, or even two different parameter configurations of the same base learner [32].

Next, we present our co-training approach for spatial data disaggregation. Then, we describe the regression methods that were selected, and how we performed the training of the models.

3.1 Co-training

Figure 1 presents our co-training framework for spatial data disaggregation. It alternates between two different regression algorithms, in order to refine initial estimates produced by a disaggregation heuristic. In more detail, the different steps are as follows.

(1) Produce a vector polygon layer for the aggregated information, by associating the source counts of the variable that is to be disaggregated with the corresponding regions;
(2) Compute initial estimates using a simple disaggregation heuristic, such as pycnophylactic interpolation [30] or dasymetric mapping proportional to population density, from the layer with source region counts produced in Step 1;
(3) Iteratively refine results using a co-training approach that leverages intermediate estimates computed at each iteration to inform the training of regression models. In odd iterations, regression model 1 uses the estimates produced by regression model 2 (or the initial estimates from the disaggregation heuristic, in the first iteration) as the regression target. In even iterations, regression model 2 instead leverages the estimates from regression model 1 as the regression target;
(4) Regardless of the regression model being used, the new estimates are adjusted for mass-preservation;
(5) Steps 3 and 4 are repeated until reaching a maximum number of iterations, or until some other stopping criteria is met.

3.2 Regression Algorithms

The selection of the regression algorithms to use in the co-training framework is crucial for the disaggregation performance. Previous work has emphasized that CNN models can achieve the best disaggregation results, although fast regression models such as random forests are very competitive and can obtain the best results in metrics such as the RMSE [22]. Taking this into account, we experimented with these two models in our co-training approach.

Specifically, random forests combine multiple decision trees, each corresponding to a non-linear procedure based on inferring a flowchart-like structure, where each internal node denotes a test on an attribute, each branch represents the outcome of a test, and each leaf node holds a target value. Decision trees can be learned by splitting

the source set of training instances into subsets, based on finding an attribute value test that optimizes the homogeneity of the target variable within the resulting subsets (e.g., by optimizing a metric such as the mean squared error). This process is recursively applied to each derived subset. The random forest approach averages the results of different decision trees that are fitted on random subsets of the features and instances, this way improving accuracy and controlling model overfitting [7].

The random forest approach processes each raster grid cell independently of the others. Since this can be limiting, we complemented it with the use of a model that takes as input patches of raster cells. We specifically use a CNN model based on the fully-convolutional U-Net architecture, similar to those adopted in studies processing remote sensing data for land coverage classification and/or image segmentation [27]. U-Nets are built upon an encoder-decoder structure, in which a contraction path (i.e., the encoder) corresponds to a stack of standard convolutional and max-pooling layers which progressively augment the number of feature maps while reducing the spatial dimensionality of the intermediate representations. The expansion path (i.e., the decoder) is then used to upscale the representations learned by the encoder, leveraging up-convolutions. Figure 2 presents a graphical representation for the U-Net architecture that was used in our work. We specifically used the architecture outlined in a previous work [22], which adds a shortcut (skip) connection to the original U-Net architecture between the final block in the expansive path and the input patches, in order to take direct advantage of the input ancillary variables.

3.3 Model Training

For the random forest model, the training procedure considered the mean squared error for measuring the quality of the splits, as well as the remaining default parameters from the scikit-learn package that supported our implementation.

In the case of the CNN model, the training procedure optimizes a multi-component loss function that explores different characteristics of the geospatial input data, namely the fact that (i) the absolute orientation of a patch of ancillary data, used in the disaggregation approach, should not affect the resulting counts, and (ii) the produced results should be spatially heterogeneous.

Specifically, we predict two outputs using our CNN model. The first, named $\hat{y}_1$, results from applying a forward pass with the model over the original patches with ancillary data. For the second, named $\hat{y}_2$, we first apply a random geometric transformation to the input patches, such as flipping over an axis or a rotation. This is followed by a forward pass with the model over the transformed patches, and finally by the application of the inverse transformation to the corresponding transformed result.

The global loss function is presented in Equation 1, and it is comprised of three components, namely:

- $\mathcal{L}\text{Cellwise}(y, \text{avg}(\hat{y}_1, \hat{y}_2))$ corresponds to an aggregated difference between the target values y and the average of the two predicted outputs $\hat{y}_1$ and $\hat{y}_2$, at the level of individual cells within the patches;
- $\mathcal{L}\text{Compatibility}(\hat{y}_1, \hat{y}_2)$ corresponds to an aggregated difference between the two versions of the predicted patches, namely between $\hat{y}_1$ and $\hat{y}_2$;

Table 1: The datasets used in our experimental evaluation.

Dataset	Source	Year	Resolution	Type
Withdrawals	National Institute of Statistics (INE)	2019	Municipalities	Aggregated
Terrain development	Global Human Settlement project	2015	38 × 38 m	Ancillary
Population density	Global Human Settlement project	2015	250 × 250 m	Ancillary
Land coverage	Copernicus Land Monitoring Service	2018	100 × 100 m	Ancillary
Human settlements	Copernicus Land Monitoring Service	2015	10 × 10 m	Ancillary
Nighttime lights	VIIRS Nighttime Lights dataset	2016	450 × 450 m	Ancillary

- $\mathcal{L}\text{Homogeneity}(\hat{y}_1, \hat{y}_2)$ is inversely derived from the standard deviation of the predictions from the CNN, which result from averaging the two output patches (i.e., $\text{std}(\text{avg}(\hat{y}_1, \hat{y}_2))$).

$$\begin{aligned}\mathcal{L}\text{Global}(y, \hat{y}_1, \hat{y}_2) = & w_1 \mathcal{L}\text{Cellwise}(y, \text{avg}(\hat{y}_1, \hat{y}_2)) + \\ & w_2 \mathcal{L}\text{Compatibility}(\hat{y}_1, \hat{y}_2) + \\ & w_3 \mathcal{L}\text{Homogeneity}(\hat{y}_1, \hat{y}_2).\end{aligned} \tag{1}$$

The three components are weighted by the parameters w_1, w_2, and w_3, which control their relative importance. Predictions from outside the region of interest are ignored, by masking out their values in the different computations of the loss.

In terms of the individual loss functions (i.e., the aggregated differences) for the components $\mathcal{L}$Cellwise and $\mathcal{L}$Compatibility, we used the standard Huber loss function defined in Equation 2, which has a quadratic behavior when the error is below a threshold δ, and a linear behaviour otherwise.

$$\text{HuberLoss}_\delta(y, \hat{y}) = \begin{cases} \frac{1}{2}(y - \hat{y})^2, & \text{for } |y - \hat{y}| \leq \delta \\ \delta(|y - \hat{y}| - \frac{1}{2}\delta), & \text{otherwise.} \end{cases} \tag{2}$$

4 EXPERIMENTAL SETUP

We evaluated the proposed spatial data disaggregation procedure through a set of experiments using socio-demographic data for the territory of Continental Portugal. We specifically considered data originally available at the level of large territorial divisions, and disaggregated it into a raster grid with a resolution of 200 meters.

Given that we lack examples of disaggregated data available at the target resolution, we relied on an evaluation strategy that compares results at the level of intermediary regions. For that, we first collect the aggregated data at a coarse level (i.e., NUTS III). We then use our disaggregation approach to estimate results for a regular raster grid with a resolution of 200 meters per cell. Finally, we re-aggregate the estimates, and compare them with known data available at intermediary regions (i.e., municipalities).

All the experiments reported in this article result from computing a fixed number of iterations (i.e., 30) of the corresponding co-training/self-training procedures. Also, in order to alleviate problems with random initializations, the values reported for all experiments using CNN models result from averaging five tests.

4.1 Datasets

Table 1 presents the aggregated data that was collected at the Portuguese National Institute of Statistics (INE), specifically containing information on the overall amount of money corresponding to withdrawals from automated teller machines, in thousands of euros, between January and December 2019.

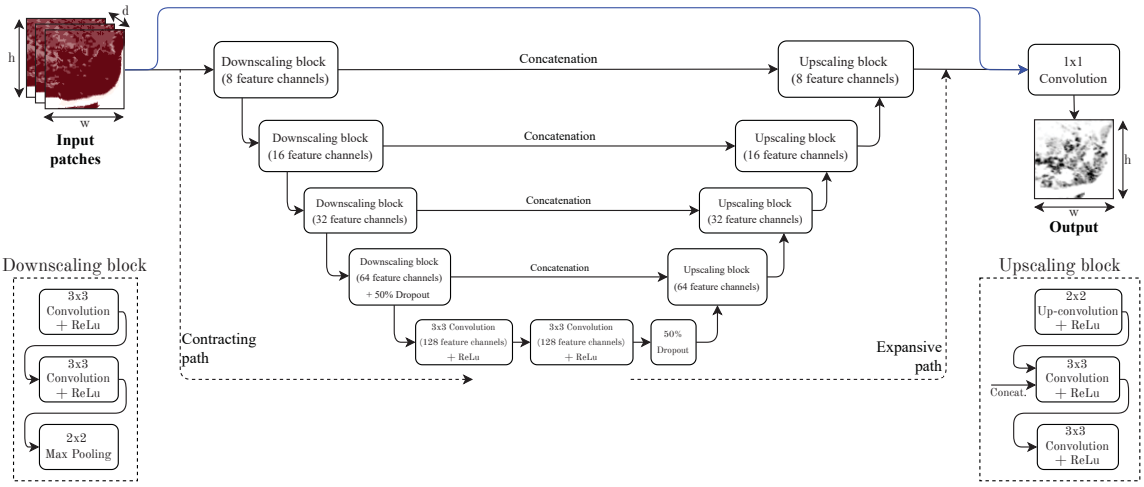

Figure 2: The encoder-decoder CNN architecture.

The ancillary information used to re-distribute the aggregated data is also presented in Table 1. We specifically collected two datasets from the Global Human Settlement[1] (GHS) project [10, 13, 14, 25]. The first concerns information on terrain development, and aims to describe the built-up structures in terms of their location and density. In our work, we used the GHS built-up presence grid related to the year of 2015, made available at a resolution of 38 meters per cell, which specifically encodes the distribution of built-up areas as the proportion of occupied footprint in each cell. The second dataset collected from the GHS project corresponds to a population density grid for the same year of 2015, with a resolution of 250 meters per cell, and was created with basis on a raster-based dasymetric mapping methodology. It specifically uses the GHS built-up presence dataset to refine the population information available through the Gridded Population of the World[2] (GPW) dataset.

We also used ancillary information collected from the Copernicus Land Monitoring Service. This included the standard Corine Landcover [3] (CLC) product, which is based on the processing of satellite images as the primary source of information [18]. We used the dataset for the year of 2018, available at a resolution of 100 meters, and converted the 44 different classes of the 3-level Corine nomenclature, considered in the original product (e.g., classes for water bodies, artificial surfaces, agricultural areas, etc.), into a real value in the range $[0, 1]$ that encodes terrain development. We also used a modern pan-European[4] dataset, made available at a spatial resolution of 10 meters per cell. This dataset represents the percentage of built-up area coverage per spatial unit, based on SPOT5 and SPOT6 satellite imagery from the year of 2015. [12].

Finally, we also considered information regarding night-time light emissions. We used the publicly available VIIRS Nighttime Lights-2016[5] dataset, which is maintained by the Earth Observation Group of the NOAA National Geophysical Data Center. We specifically selected the global cloud-free composite of VIIRS night-time lights, that was generated with VIIRS day/night band (DNB) observations collected on nights with zero moonlight. The raster data which we used, available at a resolution of 450 meters per cell, consist of floating-point values calculated by averaging the pixels deemed to be cloud-free.

Independently of their original resolution, all the datasets corresponding to ancillary variables were first converted into a resolution of 200 meters per cell, through simple upscaling/downscaling procedures such as nearest neighbour interpolation.

4.2 Evaluation Metrics

With the strategy based on intermediary aggregation zones, results can be summarized by various statistics that capture the quality of the disaggregation results, such as the Root Mean Squared Error (RMSE), the Mean Absolute Error (MAE), or the Coefficient of Determination (R^2). The corresponding formulas are as follows.

$$\mathrm{MAE}(y, \hat{y}) = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n}. \tag{3}$$

$$\mathrm{RMSE}(y, \hat{y}) = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}}. \tag{4}$$

$$\mathrm{R}^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}. \tag{5}$$

In Equations 3, 4, and 5, y_i corresponds to a ground-truth value, $\hat{y}_i$ corresponds to a predicted value, and n is the number of evaluation regions. Multiple metrics can provide a better picture of the error distribution. The MAE gives the same weight to all errors. On the other hand, the RMSE penalizes variance, as it gives errors with larger absolute values more weight than errors with smaller absolute values. The coefficient of determination R^2 measures the proportion of total variation in the ground-truth values that is explained by the model.

[1] http://ghsl.jrc.ec.europa.eu/datasets.php
[2] http://beta.sedac.ciesin.columbia.edu/data/collection/gpw-v4
[3] http://land.copernicus.eu/pan-european/corine-land-cover
[4] http://land.copernicus.eu/pan-european/GHSL/european-settlement-map
[5] http://ngdc.noaa.gov/eog/viirs/download_dnb_composites.html

4.3 Implementation Details

The entire procedure was implemented in the Python language, using frameworks such as scikit-learn[6] and Tensorflow[7]. The source code is available on a GitHub repository[8].

We tuned the hyper-parameters of the models for optimal performance, through an initial set of tests. The encoder-decoder CNN takes input patches of 16×16 grid cells (i.e., 256 cells in total). We used the Adam optimization algorithm, and set the learning rate to 10^{-3}. The δ value of the Huber loss was set to 1, and w_3 from Equation 1 was set to 40, while w_1 and w_2 were set to 1.

Taking inspiration on recent work from our team, we used a smooth weighted interpolation method for model initialization, i.e., for producing the initial estimates that are later iteratively refined [22]. This disaggregation heuristic first produces results proportional to the population distribution, which are then smoothed using one iteration of the pycnophylactic interpolation procedure.

5 EXPERIMENTAL RESULTS

We now summarize the results obtained using the co-training framework to disaggregate the spatially aggregated data mentioned in Table 1. We start by showing, in Subsection 5.1, the disaggregation errors associated with the proposed approach. We then present, in Subsection 5.2, the impact of replacing the initial estimates, as well as replacing the loss function of the CNN. We finally discuss, in Subsection 5.3, the evolution of the error across different iterations of the co-training approach, supporting our discussion on visual inspection of the results, or in terms of MAE/RMSE values.

We report, in each table that presents results, the disaggregation quality through metrics such as MAE, RMSE, and R^2. We present their values, as well as the percentage of gain over the best baseline, i.e., the smooth weighted interpolation. Tables 3 and 4 also show the standard deviation associated with the five tests computed for each experiment. The reported results correspond to the best iteration, inferred by using a stopping criterion based on the standard deviation of the resulting map. Our previous work has shown that, when dealing with spatial disaggregation techniques, one should expect the produced estimates to have high spatial heterogeneity [22]. We also concluded that the standard deviation of the results can be used as a proxy for inferring disaggregation performance, since the best iterations are usually the ones whose disaggregated map has the highest standard deviation in the target values.

5.1 Results with the Proposed Approach

In Table 2, we first present results obtained with seminal disaggregation heuristics, namely using mass-preserving areal weighting, pycnophylactic interpolation, weighted interpolation based on the population distribution, and the smooth version of the weighted interpolation (i.e., smooth weighted interpolation). We then report the results achieved with the application of a version of our self-supervised approach which only uses a single regression model. This can be seen as a standard self-training (ST) technique, similar to that of our previous work [22], and we report results when using a random forest model, as well as the CNN.

[6]http://scikit-learn.org
[7]http://www.tensorflow.org
[8]http://github.com/joaomigl15/spdisaggregation

Table 2: Results obtained with different disaggregation methods for the withdrawals on automated teller machines.

				Gain(%) / Baseline		
	MAE	**RMSE**	**R^2**	**MAE**	**RMSE**	**R^2**
Areal weighting	79667.9	205045.5	-0.0471	-368.1	-224.5	-105.2
Pycno. interpolation	78290.1	201420.9	-0.0104	-360.0	-218.8	-101.2
Weighted interpolation	17089.6	63394.2	0.8999	-0.4	-0.3	-0.1
Smooth weigh. interpolation	*17019.0*	*63181.8*	*0.9006*	—	—	—
ST w/ Random Forest (RF)	14361.2	54692.0	0.9255	15.6	**13.4**	**2.8**
ST w/ CNN	14084.4	56374.2	0.9208	**17.2**	10.8	2.2

Table 3: Results obtained with the co-training procedure for disaggregating withdrawals on automated teller machines.

				Gain(%) / Baseline		
	MAE	**RMSE**	**R^2**	**MAE**	**RMSE**	**R^2**
ST w/ CNN	14084.4±15.8	56374.2±99.0	0.9208±0.0003	17.2	10.8	2.2
CoT w/ 2 CNNs	14063.6±42.1	55941.7±357.0	0.9221±0.0010	17.4	11.5	2.4
CoT w/ CNN and RF	13906.8±54.6	54366.1±635.4	0.9264±0.0017	**18.3**	**14.0**	**2.9**
ST w/ avg. of CNN and RF	14106.3±43.4	55003.2±295.2	0.9246±0.0008	17.1	12.9	2.7

From the table, one can conclude that self-training with either of the regression models clearly outperforms the four disaggregation heuristics. Also, each of the different regressors can achieve the best results, depending on the error metric being considered. For example, self-training with the CNN leads to better MAE values, while the random forest model achieved the best RMSE and R^2.

We then assessed the application of co-training (CoT) using different regression algorithms. Table 3 presents the results in two different scenarios, namely (i) when alternating between two similar CNN models which only differ in their initializations, and (ii) when alternating between the CNN and the random forest model. We compare the results against the use of the corresponding self-training technique. We also present results with a typical ensemble approach using the same models (i.e., the CNN and the random forest model), specifically by averaging the results computed by both algorithms at each iteration, instead of alternating between them (i.e., training both models simultaneously, at each iteration, and then averaging their results).

From Tables 2 and 3, we can infer the benefit of combining the two regression algorithms (i.e., random forests and the CNN) using co-training, since it achieved the best disaggregation results in all error metrics. When comparing against self-training using the corresponding models, one can notice the increase in gains from 15.6% to 18.3% in MAE (against self-training with the random forest model), as well as from 10.8% to 14.0% in RMSE (against self-training using the CNN). It is also worth noticing that co-training using similar models (i.e., using two CNNs differing only in their initializations) had a better disaggregation performance, in all metrics, than self-training with the CNN. On the other hand, the ensemble approach that averaged the results from the random forests and the CNN produced higher errors than the co-training strategy.

Besides the improvement in disaggregation quality, the difference in training time associated with the co-training approach is also significant, when compared with self-training the CNN model. Figure 3 illustrates this, by plotting the average time associated

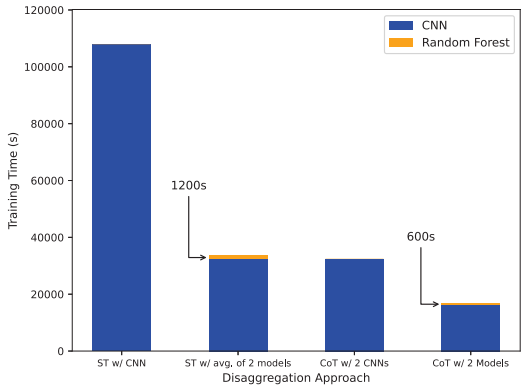

Figure 3: Training time associated with different spatial data disaggregation approaches.

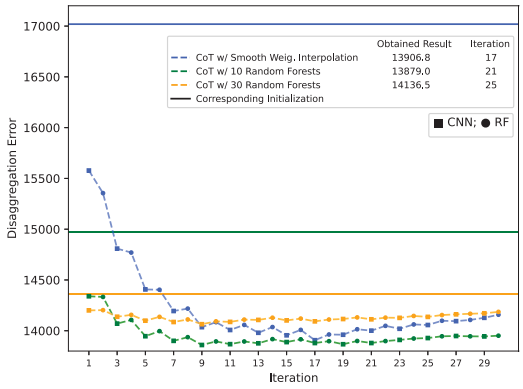

Figure 4: Disaggregation quality, measured in terms of the mean absolute error (MAE), using different approaches for the initialization of the co-training procedure.

with the execution of the experiments from Table 3. It specifically considers the average time associated with running the CNN (in blue) and the random forest model (in orange).

5.2 Result Analysis

We also explored how the co-training approach is affected when replacing two core components of the general framework, namely the approach used for computing the initial estimates, and the loss function for training the CNN. We specifically aimed to analyze how robust the method is to the presence of outliers. For that, we experimented with replacing the initial estimates, as well as the loss function used for training the CNN, with alternatives that can perhaps better focus on outlier detection and filtering.

We start by illustrating, in Figure 4, the errors obtained across 30 iterations of the co-training procedure, when leveraging different alternatives for computing the initial estimates (i.e., iteration 0). We specifically tested initial estimates resulting from separately running a self-training technique that leverages a single random forest model (i.e., similar to the results from Table 2). We collected the results from that separate procedure after 10 and 30 iterations. We then compare the error evolution of the co-training approach when using as initial estimates (i) the smooth weighted interpolation, as in previous experiments, (ii) the result of self-training using the random forest model after 10 iterations, and (iii) the result of self-training using the random forest model after 30 iterations. We specifically present the mean absolute error (MAE) of the approach used as initial estimate, as well as the same metric for each iteration of the co-training procedure.

From Figure 4, we conclude that the results from self-training using the random forest model are useful to be used as initial estimates, since they led to the best co-training results when considering only 10 iterations. More iterations (i.e., 30 iterations for training the random forest model) imposed more difficulties in generalizing towards better estimates, and led to a worse co-training performance.

To further assess how robust our approach is to the presence of outliers, Table 4 reports tests with a simpler loss function (i.e., the loss in the first two components of Equation 1) that is also sensible to their presence, namely the MSE. We also experimented

Table 4: Results obtained with different loss functions for disaggregating the withdrawals on automated teller machines.

				Gain(%) / Baseline		
	MAE	RMSE	R^2	MAE	RMSE	R^2
CoT w/ MSE	14226.6±54.4	53793.7±290.3	0.9279±0.0008	16.4	**14.9**	**3.0**
CoT w/ Huber Loss	13906.8±54.6	54366.1±635.4	0.9264±0.0017	**18.3**	14.0	2.9
CoT w/ Robust Loss	13947.2±47.3	54491.2±242.5	0.9260±0.0007	18.0	13.8	2.8

with a recently proposed robust loss function that generalizes upon different common alternatives. We specifically used the function from Barron and Jonathan [4], which generalizes several other robust loss functions. The function includes two parameters which control the overall robustness, but an adaptive version of the loss function, which we used in our experiments, can specifically learn its own parameters from training data. For comparison, we also present the result with the Huber loss (i.e, the same from Table 3). From the table, we can conclude that the MSE loss led to better results in terms of the RMSE metric, which may indicate that it was better for dealing with outliers in the data. However, it worsened the results in terms of MAE. The Huber Loss offers a good tradeoff between the different metrics, while the use of a more recent robust loss resulted in slightly worse results over all metrics.

5.3 Convergence of Self-Supervision

We analyzed the evolution of the error in our co-training approach, in terms of the impact associated with each distinct regression algorithm. In the plot from Figure 5, we can see the evolution of the results associated with two different error metrics, namely the MAE and the RMSE, when using co-training with the CNN (leveraging the Huber loss) and the random forest model. In both cases, the values are transformed using a min-max normalization, and result from applying a different algorithm at each iteration. The figure shows that co-training seems to be penalized, in terms of MAE, by the random forests. However, the RMSE metric improves at each random forest iteration, and this is perhaps linked to the best results overall (i.e., in terms of MAE and RMSE) that co-training ends up obtaining, when compared to the individual models alone.

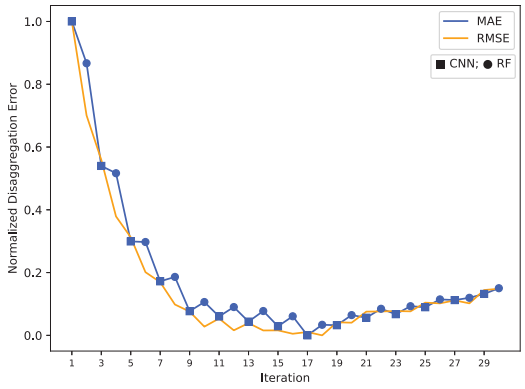

Figure 5: Disaggregation error measured in terms of the normalized MAE and RMSE.

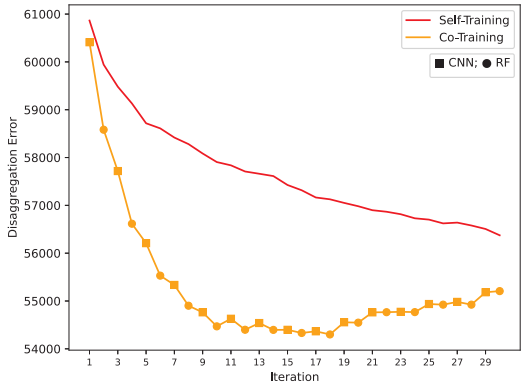

Figure 6: Disaggregation error, measured in terms of the RMSE, when using the CNN regressor in the self-training and co-training approaches.

In Figure 6, we further validate our co-training approach, by comparing the evolution of the corresponding RMSE error values, against the ones obtained when using self-training with the CNN. The figure highlights that both self-supervised techniques are useful for error reduction, since better results are obtained after Iteration 1, although co-training is particularly suitable for that purpose.

Figure 7 shows the residuals for the Portuguese municipalities within the district of Lisbon and its outskirts. It specifically maps (in the upper plot) the residuals associated with Iteration 1, and it shows (in the bottom plot) the evolution of the residuals during 4 iterations, for three municipalities highlighted in the map. Notice that odd iterations correspond to the application of the CNN, while even iterations relate to the use of the random forest model.

From Figure 7, we can observe that errors in municipalities such as *Mafra* seem to be reduced when applying the CNN, while *Odivelas* improves in the random forest iterations. Some municipalities have different types of penalizations when using different algorithms, such as *Cascais* (i.e., it is underestimated in CNN iterations, and overestimated in random forest iterations). This is perhaps

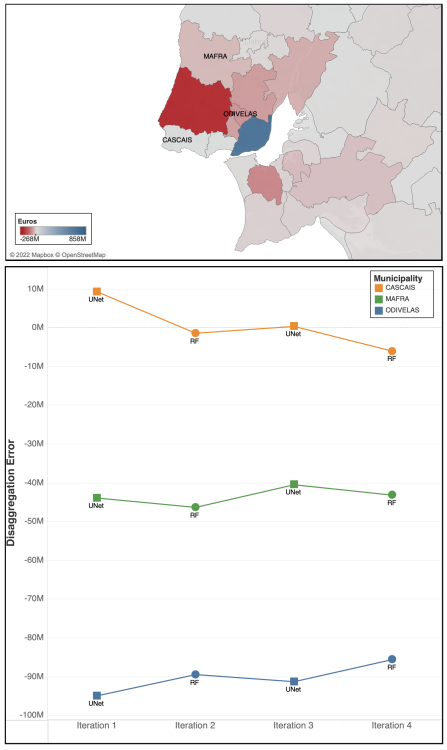

Figure 7: Residuals for the municipalities in the district of Lisbon and its outskirts.

linked to the fact that the random forest algorithm can better deal with population outliers/hotspots concentrated on one or a small number of cells (i.e., more likely present in the case of *Odivelas*, oppositely to the more rural city of *Mafra*).

6 CONCLUSIONS AND FUTURE WORK

This article reported on experiments with a spatial disaggregation technique that uses self-supervision for refining initial estimates produced through a simple disaggregation heuristic. We specifically proposed a co-training approach that alternates between two regression algorithms, namely a random forest model and an encoder-decoder CNN. We used our approach for disaggregating socio-demographic data concerning the territory of Continental Portugal, and produce estimates at a 200 meter resolution.

Different regression models can be used with the proposed method. Models based on ensembles of decision trees are an interesting choice, given that they are fast (e.g., one can use them on very large spatial regions), and can naturally deal with different types of features. For instance, these models tend to have a good performance in the presence of un-normalized data, or when dealing with very different/biased distributions. On the other hand, convolutional neural networks can achieve better results at the expense of additional computational complexity, given that they can naturally account for relations between neighboring raster cells.

The experimental results indicate that co-training with the two different regression models outperformed alternative methods based

on seminal disaggregation heuristics, as well as self-training approaches that consider a single regression model. Our method achieved the best results in all error metrics, improving over the best baseline (i.e., a smooth weighted interpolation that disaggregates the data proportionally to the population density) with 18.3% in MAE, and 14.0% in RMSE. When comparing against the application of self-training using a single model, the percentage of gains when using co-training increased from 15.6% to 18.3%, in MAE (against self-training with the random forest model), and from 10.8% to 14.0%, in RMSE (against self-training with the CNN).

Despite the interesting results, there are also many open challenges for future work in the area. The approach reported in this article can for instance be extended to consider different regression algorithms. We plan to experiment with more advanced CNN or Transformer-based architectures, and we also plan to explore different strategies to encode geospatial positioning information directly into random forests or CNN models [5, 17].

In terms of the evaluation methodology, future work can also consider some alternatives to the strategy that re-aggregates the disaggregated results to intermediary regions. One can further validate the proposed approach in a variable computed from point databases, for instance from volunteered sources such as OpenStreetMap. Aggregated information can thus be available at different levels, by summing up all the points that fall within each aggregated region. By doing so, the data computed at the coarser resolution can be disaggregated to the same 200 meter resolution, but later evaluated for smaller extents, for example cells with a resolution of 1 kilometer.

ACKNOWLEDGMENTS

We gratefully acknowledge the support of NVIDIA Corporation, with the donation of the two Titan Xp GPUs used in our experiments. This work was also partially supported by Thales Portugal, through the Ph.D. scholarship of João Monteiro, by the EU H2020 research and innovation program, through the MOOD project corresponding to the grant agreement No. 874850, and by the Fundação para a Ciência e Tecnologia (FCT), under the MIMU project with reference PTDC/CCI-CIF/32607/2017 and also under the INESC-ID multi-annual funding (UIDB/50021/2020).

REFERENCES

[1] Steven Abney. 2002. Bootstrapping. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.
[2] Maria-Florina Balcan, Avrim Blum, and Ke Yang. 2005. Co-training and Expansion: Towards Bridging Theory and Practice. *Proceedings of the Annual Meeting on Neural Information Processing Systems*.
[3] Robert E. Banfield, Lawrence O. Hall, Kevin W. Bowyer, and W. Philip Kegelmeyer. 2007. A Comparison of Decision Tree Ensemble Creation Techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29, 1.
[4] Jonathan T Barron. 2019. A General and Adaptive Robust Loss Function. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*.
[5] Thorsten Behrens, Karsten Schmidt, Raphael A Viscarra Rossel, Philipp Gries, Thomas Scholten, and Robert A MacMillan. 2018. Spatial Modelling with Euclidean Distance Fields and Machine Learning. *European Journal of Soil Science* 69, 5.
[6] Avrim Blum and Tom Mitchell. 1998. Combining Labeled and Unlabeled Data with Co-training. In *Proceedings of the International Conference on Computational Learning Theory*.
[7] Leo Breiman. 2001. Random Forests. *Machine Learning* 45.
[8] David J. Briggs, John Gulliver, Daniela Fecht, and Danielle M. Vienneau. 2007. Dasymetric Modelling of Small-area Population Distribution Using Land Cover and Light Emissions Data. *Remote Sensing of Environment* 108, 4.
[9] Zhifeng Cheng, Jianghao Wang, and Yong Ge. 2020. Mapping Monthly Population Distribution and Variation at 1-km Resolution across China. *International Journal of Geographical Information Science* 1, 1.
[10] Christina Corbane, Martino Pesaresi, Panagiotis Politis, Vasileios Syrris, Aneta J Florczyk, Pierre Soille, Luca Maffenini, Armin Burger, Veselin Vasilev, Dario Rodriguez, et al. 2017. Big Earth Data Analytics on Sentinel-1 and LandSat Imagery in Support to Global Human Settlements Mapping. *Big Earth Data* 1.
[11] Erin Doxsey-Whitfield, Kytt MacManus, Susana B Adamo, Linda Pistolesi, John Squires, Olena Borkovska, and Sandra R Baptista. 2015. Taking Advantage of the Improved Availability of Census Data: A First Look at the Gridded Population of the World, Version 4. *Papers in Applied Geography* 1, 3.
[12] Aneta Jadwiga Florczyk, Stefano Ferri, Vasileios Syrris, Thomas Kemper, Matina Halkia, Pierre Soille, and Martino Pesaresi. 2016. A New European Settlement Map from Optical Remotely Sensed Data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 9, 5.
[13] Sergio Freire, Erin Doxsey-Whitfield, Kytt MacManus, Jane Mills, and Martino Pesaresi. 2016. Development of New Open and Free Multi-temporal Global Population Grids at 250m Resolution. In *Proceedings of the AGILE International Conference on Geographic Information Science*.
[14] Sergio Freire, Thomas Kemper, Martino Pesaresi, Aneta Florczyk, and Vasileios Syrris. 2015. Combining GHSL and GPW to Improve Global Population Mapping. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*.
[15] Francisco Javier Gallego. 2010. A Population Density Grid of the European Union. *Population and Environment* 31, 6.
[16] Michael F. Goodchild, Luc Anselin, and Uwe Deichmann. 1993. A Framework for the Areal Interpolation of Socioeconomic Data. *Environment and Planning A* 25, 3.
[17] Tomislav Hengl, Madlene Nussbaum, Marvin N Wright, Gerard BM Heuvelink, and Benedikt Gräler. 2018. Random Forest as a Generic Framework for Predictive Modeling of Spatial and Spatio-temporal Variables. *PeerJ* 6.
[18] Y. Heymann, C. Steenmans, G. Croisille, and M. Bossard. 1994. *CORINE Land Cover Technical Guide*. Technical Report. Office for Official Publications of the European Communities.
[19] Nathan Jacobs, Adam Kraft, Muhammad Usman Rafique, and Ranti Dev Sharma. 2018. A Weakly Supervised Approach for Estimating Spatial Density Functions from High-resolution Satellite Imagery. In *Proceedings of the ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*.
[20] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based Learning Applied to Document Recognition. *Proceedings of the IEEE* 86, 11.
[21] Brendan P. Malone, Alex B. McBratney, Budiman Minasny, and Ichsani Wheeler. 2012. A General Method for Downscaling Earth Resource Information. *Computers and Geosciences* 41, 1.
[22] João Monteiro, Bruno Martins, Miguel Costa, and João M Pires. 2021. Geospatial Data Disaggregation through Self-Trained Encoder–Decoder Convolutional Models. *ISPRS International Journal of Geo-Information* 10, 9.
[23] João Monteiro, Bruno Martins, Patricia Murrieta-Flores, and João M Pires. 2019. Spatial Disaggregation of Historical Census Data Leveraging Multiple Sources of Ancillary Information. *ISPRS International Journal of Geo-Information* 8, 8.
[24] João Monteiro, Bruno Martins, and João M Pires. 2018. A Hybrid Approach for the Spatial Disaggregation of Socio-economic Indicators. *International Journal of Data Science and Analytics* 5, 2-3.
[25] Martino Pesaresi, Daniele Ehrlich, Stefano Ferri, Aneta Florczyk, Sergio Freire, Matina Halkia, Andreea Julea, Thomas Kemper, Pierre Soille, and Vasileios Syrris. 2016. *Operating Procedure for the Production of the Global Human Settlement Layer from LandSat Data of the Epochs 1975, 1990, 2000, and 2014*. Technical Report. Publications Office of the European Union.
[26] Ge Qiu, Yuhai Bao, Xuchao Yang, Chen Wang, Tingting Ye, Alfred Stein, and Peng Jia. 2020. Local Population Mapping using a Random Forest Model Based on Remote and Social Sensing data: A Case Study in Zhengzhou, China. *International Journal of Remote sensing* 12, 10.
[27] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*.
[28] Forrest R. Stevens, Andrea E. Gaughan, Catherine Linard, and Andrew J. Tatem. 2015. Disaggregating Census Data for Population Mapping Using Random Forests with Remotely-sensed and Ancillary Data. *PloS one* 10, 2.
[29] Tobias G Tiecke, Xianming Liu, Amy Zhang, Andreas Gros, Nan Li, Gregory Yetman, Talip Kilic, Siobhan Murray, Brian Blankespoor, Espen B Prydz, et al. 2017. Mapping the World Population One Building at a Time. *arXiv preprint arXiv:1712.05839* (2017).
[30] Waldo R. Tobler. 1979. Smooth Pycnophylactic Interpolation for Geographical Regions. *Journal of the American Statistical Association* 74, 367.
[31] Wei Wang and Zhi-Hua Zhou. 2007. Analyzing Co-training Style Algorithms. In *Proceedings of the European Conference on Machine Learning*.
[32] Zhi-Hua Zhou, Ming Li, et al. 2005. Semi-supervised Regression with Co-training. In *Proceedings of the International Joint Conference on Artificial Intelligence*.