Lost and Found: How Self-Supervised Learning Helps GPS Coordinates Find Their Way

Nicolas Bougie NICOLAS.BOUGIE@WOVEN.TOYOTA
Daria Vazhenina DARIA.VAZHENINA@WOVEN.TOYOTA
Narimasa Watanabe NARIMASA.WATANABE@WOVEN.TOYOTA
Woven by Toyota, Tokyo, Japan

Editors: Berrin Yanıkoğlu and Wray Buntine

Abstract

GPS coordinates are a fundamental aspect of location-based applications, yet prior methods for representing them do not fully capture the intricate relationships between different locations. In this paper, we propose a novel map-based approach to embedding GPS coordinates using self-supervised learning. Unlike most prior studies that directly embed GPS coordinates to a latent space, we leverage a map-based approach, allowing embeddings to capture geographical and economic features. Namely, we use a student-teacher architecture, where a student network is trained to mimic the outputs of the teacher, using two different augmented versions of the same input. To capture the rich underlying semantics of GPS coordinates, we further leverage auxiliary tasks including *geo* prediction, high-level reconstruction, and intermediate clustering. The intermediate clustering loss facilitates learning features at different levels of granularity, while the high-level reconstruction loss encourages "local-to-global" correspondences. We evaluate our method on a large-scale dataset of GPS coordinates and demonstrate that it outperforms several baseline methods in terms of the quality of the learned embeddings. Moreover, we show the usefulness of our embeddings in various downstream tasks, such as predicting land price, land cover type, or water quality indice.

Keywords: Self-Supervised Learning; GPS embedding; Machine Learning

1. Introduction

Location-based applications Li et al. (2019) leverage GPS coordinates as a foundational element, empowering a plethora of services and functionalities, spanning from navigation Herrera et al. (2010) and ride-sharing to geotagging and beyond Raper et al. (2007). GPS-based applications also hold substantial promise in shaping the landscape of smart cities Wov (2023). These applications cover a wide range of areas in smart cities, including location-based services and understanding human behavior from GPS traces Wang (2016). However, traditional techniques for representing GPS coordinates, such as latitude and longitude, provide only a limited representation of location and fail to capture the complex relationships between different locations Jean et al. (2019). For instance, two locations with similar latitude and longitude values may have vastly different geographic or economic features, making it challenging to develop effective applications that rely on the semantic meaning of locations. To address this limitation, researchers have proposed various methods for embedding GPS coordinates into a high-dimensional space Yin et al. (2019), where a GPS coordinate is represented as a vector that captures its semantic meaning.

While prior work have explored various methods for GPS coordinate embedding, such as using convolutional neural networks (CNNs) Dabiri et al. (2020) and graph-based models Tian et al. (2021), these methods have limitations in capturing rich semantic information. For example, the work by Dabiri et al. uses a CNN to embed GPS coordinates, however, the model is limited by its inability to capture contextual information from surrounding locations. Similarly, Tian et al. Tian et al. (2021) proposed a graph-based model for GPS coordinate embedding, but it relies on a pre-defined graph structure that may not capture the nuances of real-world geographic relationships. In contrast, the proposed map-based methodology allows for the extraction of contextual information from surrounding locations, enabling the model to capture more complex relationships between locations. Namely, our approach differs from prior works that directly embed GPS coordinates, as it leverages maps to incorporate contextual information from surrounding locations. Additionally, we employ auxiliary tasks to extract rich semantic information from maps.

This paper presents a novel approach to embedding GPS coordinates into a high-dimensional space using self-supervised learning, GPS-SELM (**GPS**-coordinate **S**elf-supervised **E**mbedding with **L**ocation-based **M**aps). We leverage a map-based approach that allows embeddings to capture rich features, including geographical and economic information. Specifically, we use a student-teacher architecture, where a student network is trained to mimic the output of the teacher network, using two different views of the same input. Namely, this knowledge distillation loss encourages the student network to match the outputs of the teacher network. To incentivize the network to learn richer embeddings, we further leverage auxiliary tasks including *geo* prediction and high-level reconstruction. The high-level reconstruction task seeks to capture "local-to-global" correspondences. In addition, the model is equipped with an intermediate clustering module that pulls similar samples close. This brings similar samples close and thus pulls similar local-groups together, capturing features at a semantic group level. The resulting embeddings can be used for a variety of downstream tasks, including geographic information retrieval and clustering.

To evaluate the effectiveness of GPS-SELM, we conduct experiments on large-scale datasets of GPS coordinates. We compare the performance of our approach against several baseline methods, including prior work on GPS coordinate embedding. The experimental results demonstrate that our approach outperforms vanilla DINO Caron et al. (2021) in terms of the quality of the learned embeddings. In particular, we improve it by ~7% classification accuracy when testing the learned representations on an area-type classification task, ~5% top-1 accuracy on land cover classification, ~5% accuracy on area-type prediction, and ~1% AP on object detection. Surprisingly, our model can even outperform the remote sensing counterpart on water quality prediction using remote image sensing images by ~3% top-1 accuracy.

2. Related Work

In the quest to encode the underlying semantics of GPS coordinates, researchers have actively explored the utilization of a wide range of supplementary data sources and techniques, including self-supervised learning and extracting nuanced semantic contexts.

Self-Supervised Learning One prominent research direction is contrastive learning, which includes MoCo He et al. (2020) and MoCo-V2 Chen et al. (2020). These methods have consistently showcased superior performance across downstream tasks. The underlying principle of contrastive learning is to train representations by pulling positive image pairs from the same instance closer together in latent space while pushing negative pairs from different instances further apart. Contrary to contrastive approaches, Grill et al. (Grill et al. (2020)) introduce a metric-learning framework (BYOL) which trains features by matching them to representations obtained from a momentum encoder. While methods like BYOL can function without a momentum encoder, they experience a slight performance decline Chen and He (2021). Several other works align with this approach, demonstrating the ability to match more intricate representations Gidaris et al. (2020). DINO Caron et al. (2021) simplifies self-supervised training by directly predicting the output of a teacher network, constructed using a momentum encoder, through the utilization of a standard cross-entropy loss. MUGS Zhou et al. (2022) proposed to explicitly learn multi-granular visual features. GPS-SELM employs a similar approach for intermediate clustering while explicitly capturing "local-to-global" correspondences and map-related features. In addition, we propose a different way of creating soft cluster and use a different network architecture. Despite the rapid growth of self-supervised learning, its application on GPS embedding remains largely unexplored. Prior studies predominantly rely on small-scale datasets that are constrained to specific geographic regions Jean et al. (2019); Lu et al. (2017) or highly specialized modalities like hyperspectral images Mou et al. (2017). Although these approaches have demonstrated effectiveness in addressing challenges specific to remote sensing, their applicability to GPS embeddings is limited due to the inherent differences between the domains.

GPS Coordinates to Embeddings. Several approaches have been proposed to leverage supplementary data sources for encoding semantics onto GPS coordinates. For instance, Joshi and Luo Joshi (2008) utilized GeoNames, a publicly available geographical information system database. That is, they discriminatively model the statistical saliency of geo-tags in describing an activity or event. In a similar spirit, other researchers Tang et al. (2015) leverage Google Maps to extract geographic and statistical features for locations within the United States. In GPS2Vec Yin et al. (2019), a two-level grid-based framework is used to learn semantic embeddings for geo-coordinates worldwide. On the other hand, it may be possible to exploit the spatio-temporal structure of remote sensing data Ayush et al. (2021). The authors exploit spatially aligned images over time to construct temporal positive pairs in contrastive learning and geo-location to design pretext tasks. Another related paper Samano et al. (2020) proposed an approach to geolocalising panoramic images on a 2D cartographic map based on learning a low-dimensional embedded space. To further improve the discriminatory to allow localization, they enhance their model by concatenating along a route. Nevertheless, the current methods have notable drawbacks: 1) limited applicability at inference due to reliance on specific supplementary data sources; 2) high computational cost resulting from frequent nearest neighbor search queries; and 3) lack of economic and rich semantic information, narrowing down potential applications.

3. Method

Our method (GPS-SELM) leverages a map-based approach to embedding GPS coordinates using self-supervised learning (Figure 1). Unlike traditional methods for GPS coordinate representation, which can be limited in their ability to capture the variability of location and semantic information, our approach uses map images to capture rich geographical and topological features. By projecting each GPS coordinate onto a map image, GPS-SELM can handle a large number of

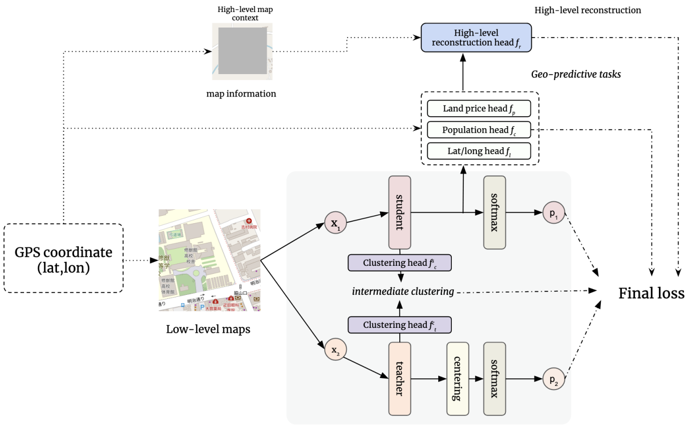

Figure 1: Overall framework of GPS-SELM. The input GPS coordinate is projected onto a map image. For each map image, GPS-SELM performs random augmentations and feeds two crops into backbones of student and teacher. Next, we leverage three types of auxiliary tasks: 1) geo-predictive tasks, 2) a high-level reconstruction task, and 3) an intermediate clustering task. The final loss is the weighted sum of the self-supervised and auxiliary losses.

entities within a region, without the need for manual feature selection or creation. To do so, we use a student-teacher with knowledge distillation.

In detail, first, each GPS coordinate is projected onto a map image centered on the input coordinate. Then, a student and teacher networks receive two different augmented versions of the same map image. The augmentations include random rotations, zooming, and crops of the input. Through knowledge distillation, the student network learns meaningful embeddings that capture geographical and topological features. We also introduce auxiliary tasks to further improve the quality of the learned embeddings, including geo-predictive tasks and a high-level reconstruction task. Auxiliary tasks primarily aim to extract economic and demographic information from map images. Finally, we equip the model with an intermediate clustering loss to encourage pulling together similar GPS coordinates at a group level.

3.1. From GPS Coordinates to Maps

As mentioned above, we introduce a map-based approach that converts GPS coordinates into map images to leverage the rich spatial and contextual information provided by maps. To generate map images, we retrieved map tiles corresponding to the GPS coordinates from OpenStreetMap OpenStreetMap contributors (2017) (OSM). We store in a dataset **D** pairs of GPS coordinates (x, y) and their associated map m, $D = \{(x_0, y_0) \rightarrow m_0, (x_1, y_1) \rightarrow m_1, \cdots\}$.

Since a large number of maps have a low semantic meaning such as *ocean* or *mountain*, we use an entropy-based cleaning strategy. We filter out maps with low entropy to remove irrelevant maps and retain only semantically meaningful maps, such as roads or buildings. Namely, we add a map to **D** if the image entropy is larger than a threshold $t_{entropy}$.

During training, we further employ prioritized sampling, where we assign higher sampling probabilities to GPS coordinates that are less represented in the dataset, ensuring a more diverse set of training examples. In particular, we assume that complex maps such as maps representing urban environments should be given more weight. Concretely, we define the probability of sampling a map m_i as:

$$P(m_i) = \frac{p_i^{\alpha}}{\sum_k p_k^{\alpha}}, \tag{1}$$

where $p_i > 0$ is the priority of map m_i. The exponent α determines how much prioritization is used, with $\alpha = 0$ corresponding to the uniform case. In the current implementation, we consider an entropy-based prioritized sampling where $p_i = \mathcal{H}(m_i) + \epsilon$, where $\mathcal{H}$ is the entropy function and ϵ is a small positive constant that prevents the edge-case of maps not being used.

3.2. Self-Supervised Learning via Student-Teacher Distillation

We use a self-supervised learning approach with student-teacher distillation, inspired by DINO Caron et al. (2021), to extract meaningful embeddings from map images. The method shares similarities with knowledge distillation Hinton et al. (2015) and DINO Caron et al. (2021).

Knowledge distillation is a learning technique in which a student network, denoted by g_{θ_s}, is trained to match the output of a teacher network, denoted by g_{θ_t}. The networks are parameterized by θ_s and θ_t, respectively. Given an input map image m, both networks output probability distributions over K dimensions represented by P_s and P_t. These probability distribution P is obtained by normalizing the output of the network g with a softmax function. The process for the student network g_{θ_s} is formalized below:

$$P_s(m)^{(i)} = \frac{\exp(g_{\theta_s}(m)^{(i)}/\tau_s)}{\sum_{k=1}^{K}\exp(g_{\theta_s}(m)^{(k)}/\tau_s)}, \tag{2}$$

where $\tau_s > 0$ is a temperature parameter that controls the sharpness of the output distribution. An analogous formula holds for P_t with temperature τ_t. Given the fixed teacher network g_{θ_t}, the student network seeks to match these distributions by minimizing the cross-entropy loss L_{cl} w.r.t. the parameters of the student network θ_s:

$$\min_{\theta_s} \mathcal{H}(P_t(m), P_s(m^{'})), \tag{3}$$

where $\mathcal{H}(a,b) = -a\log b$, and, m and $m^{'}$ are two different views of the same input map image. Given that our method utilizes map images that exhibit relatively consistent intensity and quality, we use augmentations that include random rotations, zooming, and crops of the input. The overall process can be formulated as follows:

$$\min_{\theta_s} \sum_{m\in\{m_1^g, m_2^g\}} \sum_{m'\in V \wedge m' \neq m} \mathcal{H}(P_t(m), P_s(m^{'})), \tag{4}$$

where for a given map m, we generate a set V of different views. This set contains two global views, m_1^g and m_2^g and multiple local views of smaller resolution with random augmentations. All augmented crops are passed through the student while only the global views are passed through the teacher.

The teacher network's weights are updated using an exponential moving average (EMA) on the student weights, i.e., a momentum encoder He et al. (2020). The update rule is $\theta_t \leftarrow \lambda\theta_t + (1-\lambda)\theta_s$, with λ following a cosine schedule from 0.996 to 1 during training.

3.3. Auxiliary Tasks

To further improve the quality of the learned embeddings, we introduce three types of auxiliary tasks. First, we leverage a set of *geo-predictive* tasks, including land price prediction, population density estimation, and latitude/longitude prediction. Second, we add a high-level reconstruction task where the student network is trained to reconstruct a higher-level map of the input map. This helps the student network to capture "local-to-global" correspondences. Finally, we equip the model with an intermediate clustering loss that seeks to learn group-level features. The resulting model can capture the spatial relationships between different locations and generate embeddings that encapsulate both macro and micro-level features of the map.

3.3.1. GEO-PREDICTIVE TASKS

First, we employ a land price prediction task, where the land price head f_p being attached to the student network is trained to predict the land price of the input map image. This task is based on a mean squared error (MSE) loss and is computed as follows:

$$L_{lp} = \frac{1}{N}\sum_{j=1}^{N}(y_j - \hat{y}_j)^2 = \frac{1}{N}\sum_{j=1}^{N}(y_j - f_p(z^j))^2, \tag{5}$$

where z^j is the feature vector produced by the student encoder, N is the number of training examples, y_i is the true land price of the location corresponding to the j-th example, and $\hat{y}_i$ is the predicted land price.

Second, we add a population density clustering task, where a density clustering head f_c is trained to predict the population density of the input map image. Ground-truth $\mathscr{K}$ clusters were given by a clustering method (i.e., k-means), which assigned an area with coordinates (x_i, y_i) a categorical density label $c_i \in C = \{1, \cdots, \mathscr{K}\}$. Using the cross entropy loss function, we then optimize a density clustering head f_c to recover the ground-truth density clusters as:

$$L_c = \frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{\mathscr{K}} -p(c_i = k)\log(\hat{p}(c_i = k|f_c(z^j))), \tag{6}$$

where z^j is the feature vector produced by the student encoder, N is the number of training examples, $\mathscr{K}$ is the number of classes for the population density, and c_i is the true population density class for the i-th example.

Finally, we leverage a latitude/longitude prediction task, where a *latlong* prediction head f_l is trained to predict the latitude and longitude coordinates of the input map image. This task is based on an MSE loss and is computed as follows:

$$L_l = \frac{1}{N}\sum_{i=1}^{N}\left[(lat_i - \hat{lat}_i)^2 + (lon_i - \hat{lon}_i)^2\right], \tag{7}$$

where N is the number of training examples, lat_i and lon_i are the true latitude and longitude coordinates of the location corresponding to the i-th example, and $\hat{lat}_i$ and $\hat{lon}_i$ are the predicted latitude and longitude coordinates generated by the *latlong* prediction head f_l given the feature vector produced by the student encoder z^j.

We propose the objective for joint learning as the linear combination of geo-predictive losses with uniform coefficients as:

$$\underset{\theta_s}{\operatorname{argmin}}\ L_{geo} = \frac{1}{3}L_{lp} + \frac{1}{3}L_c + \frac{1}{3}L_l. \tag{8}$$

3.3.2. High-Level Reconstruction Task

To encourage the student network to learn "local-to-global" correspondences, we introduce a high-level reconstruction task. A high-level reconstruction head f_r is trained to reconstruct higher-level features of the input, enabling the student network to capture global information about the map such as road shape, land use, and terrain. High-level map images can be retrieved from OpenStreetMap by collecting map images at a smaller zoom level. To aid the student network in the reconstruction task, we provide the *context* of the high-level image — the border pixels (see Figure 1). These border pixels are added to the input of f_r, providing *context* of the image to f_r. We use a mean squared error (MSE) loss to measure the difference between the reconstructed image and the original high-level image. Let $\hat{M}$ be a binary mask corresponding to the border region, with a value of 0 for context pixels and 1 for other pixels. The loss is computed as follows:

$$L_{rec} = \frac{1}{N}\sum_{j=1}^{N} ||(m_j^* - f_r(z^j, m_j^{**})) \odot \hat{M}||_2^2, \tag{9}$$

where m_j^* is a higher version of the original input image m_j, z^j is the feature produced by the student encoder, m_j^{**} is the context of m_j^*, and $\odot$ is the element-wise product operation.

3.3.3. Intermediate Clustering Task

In order to capture features at a different level of granularity — group level, we introduce an intermediate clustering matching task, which learns a clustering matching between the student and teacher networks. This involves clustering the intermediate embeddings generated by the student and teacher networks and matching the resulting clusters to ensure that they are similar. Clustering is performed on soft labels generated by the student and teacher networks, which are probability distributions over the clusters. By using soft labels, we can capture the uncertainty of the student network's predictions and enable the matching to be performed at a group level.

Namely, we feed the cluster token y_c^t in the feature z_t from backbone teacher and the cluster token y_c^s in z_s from student backbone into two heads f_c^t and f_c^s, where f_c^t and f_c^s are attached to the student and teacher respectively. We then construct a set of learnable cluster prototypes $\{c_i\}_{i=1}^m$ and compute soft pseudo clustering labels for the student as:

$$p_i^s = \frac{\exp((f_c^s(y_c^s) - p_{div}) \cdot c_i / \tau_c^{'})}{\sum_{i=1}^m \exp((f_c^s(y_c^s) - p_{div}) \cdot c_i / \tau_c^{'})}, \tag{10}$$

and for teacher as:

$$p_i^t = \frac{\exp((f_c^t(y_c^t) - p_{div}) \cdot c_i / \tau_c)}{\sum_{i=1}^m \exp((f_c^t(y_c^t) - p_{div}) \cdot c_i / \tau_c)}, \tag{11}$$

where the term p_{div} is used to avoid discovering the trivial solution of mapping most data points to the same cluster. p_{div} is defined as the exponential moving average of all past $f_c^t(y_t^c)$,

$p_{div} \leftarrow \eta \cdot p_{div} + (1-\eta) \cdot \frac{1}{|\mathscr{B}|} \sum_{p^t \in \mathscr{B}} p^t$, where $\mathscr{B}$ is a mini-batch of size 256 and η is a hyperparameter of our method. The model is trained to minimize a cross-entropy of soft cluster assignment probabilities defined below:

$$L_{clust}(m, m^{'}) = -\sum_{i=1}^{m} p_i^t \log(p_i^s). \tag{12}$$

3.4. Overall Training

The final loss of GPS-SELM is defined as:

$$L_{total} = L_{cl} + \alpha L_{geo} + \beta L_{rec} + \gamma L_{clust}, \tag{13}$$

where L_{cl} is the cross-entropy loss, L_{geo} is the sum of the three geo-predictive losses, L_{rec} is the high-level reconstruction loss, and L_{clust} is the intermediate clustering matching loss. α, β, and γ are coefficients to trade-off the three losses, allowing us to control the relative importance of each loss during training.

4. Results

Implementation Details. For self-supervised learning, we use ResNet-50 to parameterize the student and teacher encoders, in all experiments. We pretrain the models on map images (128×128) collected from OpenStreetMap for all Japan at zoom level 17, and high-level images for the reconstruction task at zoom level 16. We train with the AdamW optimizer Loshchilov and Hutter (2017) and a batch size of 256. Similarly to DINO Caron et al. (2021), the learning rate is linearly ramped up during the first 10 epochs and was set to lr = 0.0005. After this warmup, we decay the learning rate with a cosine schedule. The weight decay also follows a cosine schedule from 0.04 to 0.4. The temperature τ_s is set to 0.1 in Eq. 2, while we use a linear warm-up for τ_t from 0.04 to 0.07 during the first 30 epochs. We set $t_{entropy}$ to 0.40, and the projection dimension K of student/teacher networks to 256. For geo-predictive tasks, the three heads (i.e., f_p, f_c, and f_l) are all 2-layered MLPs with hidden dimension 256 and rectified linear unit (ReLU) activations. The high-level reconstruction head f_r consists of a series of 3 transposed convolutional layers with Gaussian Error Linear Unit (GELU) activation. We define the *context* of the input as the area within a 3-pixel border around the map image. For the intermediate clustering matching task, we set $\tau_c^{'} = 0.1$ (Eq. 10) and linearly warm up τ_c from 0.04 to 0.07. We selected the neighbor number k = 10, and $\eta = 0.9$ in the intermediate clustering module. The two projection heads are 3-layered MLP with hidden/output dimension of 512/256. We set loss coefficients as $\alpha = 0.3$, $\beta = 0.3$, and $\gamma = 0.4$. Area information employed in geo-predictive tasks are publicly available on the Ministry of Land, Infrastructure, Transport and Tourism (MLIT) website MLIT (2023). As described in section 3.3.1, we reduce the complexity of the population density classification problem by applying a k-mean model on population density values, reducing the number of unique classes from 588 to 30. We pretrained the model for 100 epochs on the OpenStreetMap dataset, and 50 epochs on the downstream tasks.

Baselines. We compare the proposed method with several baselines, including MoCo-V2 Chen et al. (2020), DINO Caron et al. (2021), MoCo-V2+Geo+TP Ayush et al. (2021), and Tile2Vec Jean et al. (2019). In addition, for downstream experiments we report results with 1) random initialization while fine-tuning on the target task (*Random Init*), and 2) a simple NN consisting

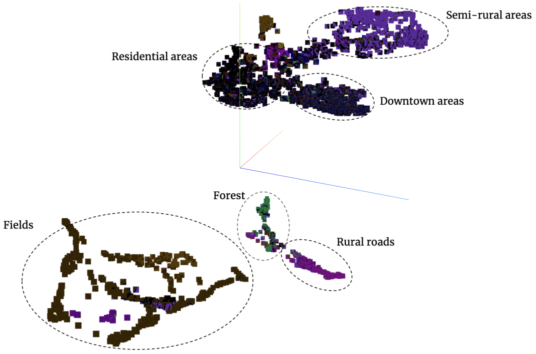

Figure 2: t-SNE visualization of feature embeddings on the OpenStreetMap dataset. Each square represents a map feature, and the dotted circles indicate the cluster "name".

of 3-layered MLP with a hidden dimension of 128 that was trained on raw GPS coordinates (*raw GPS*). We compare those baselines with three variations of GPS-SELM: 1) GPS-SELM solely trained with geo-predictive tasks *GPS-SELM (geo)*, 2) GPS-SELM (geo) augmented with the high-level reconstruction task *GPS-SELM (geo+rec)*, and 3) the full model *GPS-SELM (geo+rec+int)*.
Datasets. We conducted several of the experiments to evaluate the performance of GPS-SELM on the OpenStreetMap dataset, which consists of 8,924,351 map tiles before filtering. We used map images of size 128×128 pixels. We randomly split the dataset into training, validation, and testing sets with a ratio of 70:10:20, respectively. We also employed data from the National Agricultural Imagery Program (NAIP) DOI/USGS/EROS (2021), xView dataset Lam et al. (2018), and SustainBench dataset Yeh et al. (2021) for further analysis.

4.1. t-SNE Visualization

We applied t-SNE to visualize the embeddings produced by our method on the OpenStreetMap dataset. We randomly selected 2,048 maps from the evaluation set and extracted their corresponding feature embeddings using the trained student network. We then reduced the dimensionality of the embeddings to three dimensions using t-SNE. Figure 2 shows the t-SNE visualization of the feature embeddings. We observe that the embeddings form distinct clusters that correspond to different map features, such as buildings, roads, and vegetation. Within each cluster, we carried out additional analysis by leveraging ground-truth labels (i.e., land price, land cover, area type), and observe sub-clusters that correspond to specific subclasses of features, such as residential buildings, commercial buildings, and industrial buildings. Furthermore, we could notice that the embeddings capture the spatial relationships between map features. For example, the building clusters are located in the center of the visualization, surrounded by the road and vegetation clusters. Besides, we discovered sub-clusters of points within the urban area clusters that focused on specific features such as land price and population density of the area.

4.2. Land Price Prediction

Next, we evaluate GPS-SELM on a downstream task, land price prediction. The goal was to predict land price around the given GPS coordinate. We randomly selected 50,000 GPS coordinates and

Pretrain	**1 epoch**	**25 epochs**	**50 epochs**
Random Init	-/0.93	-/0.45	-/0.28
raw GPS	-/1.92	-/1.70	-/1.67
MUGS Zhou et al. (2022)	0.68/0.66	0.47/0.42	0.31/0.27
MoCo-V2 Chen et al. (2020)	0.67/0.64	0.39/0.36	0.30/0.25
DINO Caron et al. (2021)	0.65/0.61	0.41/0.36	0.29/0.26
MoCo-V2+Geo+TP Ayush et al. (2021)	0.58/0.53	0.45/0.40	0.28/0.24
Tile2Vec Jean et al. (2019)	0.60/0.57	0.41/0.36	0.26/0.22
GPS-SELM (geo)	0.29/0.25	0.22/0.17	0.16/0.14
GPS-SELM (geo+rec)	0.27/0.23	0.19/0.16	0.15/0.12
GPS-SELM (geo+rec+int)	**0.26/0.22**	**0.17/0.14**	**0.13/0.10**

Table 1: Experiments on GPS-SELM on predicting land price. We report (frozen/finetune) mean squared errors (MSE), averaged over ten evaluation trials. Frozen corresponds to linear classification on frozen features. Finetune corresponds to end-to-end fine-tuning results.

their maps from the OpenStreetMap dataset, and, extracted the corresponding feature embeddings using the trained student network. For a fair comparison, the map images used for the land price prediction task in the downstream task were not used for pretraining our method. Namely, we evaluated GPS-SELM on GPS coordinates from the *Hokkaido* area (i.e., north of Japan), while pretraining samples contain GPS coordinates for other parts of Japan. Table 1 presents the results of our experiments, including frozen and finetune MSE of each method. Vanilla Moco-V2 and DINO were outperformed by MoCo-V2+Geo+TP and Tile2Vec baselines. Moreover, the proposed method achieved the lowest errors of 0.22 and 0.10 after 1 epoch and after 50 epochs of training, respectively. One compelling reason for the effectiveness of GPS-SELM is its ability to transfer external knowledge such as land price seen during the pretraining, which other baselines lack. Besides, one can observe that *raw GPS* is outperformed by map-based approaches including GPS-SELM, highlighting the importance maps rather than raw GPS coordinates.

4.3. Area Type

We report the results in Table 2 on an area type classification task. Given a GPS coordinate as input, the goal is to classify the land into one of 70 predefined types, such as *industrial land*, *agricultural land*, etc. The evaluation is conducted using top-1 and top-5 accuracy metrics, averaged over ten evaluation trials. True classes were obtained via the Ministry of Land, Infrastructure, Transport and Tourism (MLIT) MLIT (2023). The experiment compares the proposed GPS-SELM method against four baseline approaches: MoCo-V2, DINO, MoCo-V2+Geo+TP, and Tile2Vec. Additionally, three variations of GPS-SELM are evaluated: GPS-SELM (geo), GPS-SELM (geo+rec), and GPS-SELM (geo+rec+int). Among the baseline methods, MoCo-V2, DINO, MoCo-V2+Geo+TP, and Tile2Vec, GPS-SELM consistently outperforms them in terms of accuracy. The results show that incorporating geographical information, reconstruction, and intermediate clustering tasks in GPS-SELM (geo+rec+int) leads to the highest accuracy values at each evaluation stage. This indicates the effectiveness of leveraging multiple components for improved area type prediction.

Pretrain	1 epoch	25 epochs	50 epochs
Random Init	19.23/23.20	24.69/35.98	31.51/44.31
raw GPS	2.45/3.98	4.16/6.12	4.09/6.10
MUGS Zhou et al. (2022)	27.42/41.47	29.51/45.18	33.75/46.90
MoCo-V2 Chen et al. (2020)	27.37/44.12	30.06/47.80	35.61/49.17
DINO Caron et al. (2021)	29.56/44.63	32.15/48.69	36.79/50.91
MoCo-V2+Geo+TP Ayush et al. (2021)	29.49/46.30	33.09/49.87	35.28/52.33
Tile2Vec Jean et al. (2019)	23.02/39.20	27.90/44.04	32.01/47.20
GPS-SELM (geo)	33.25/50.90	34.37/53.66	39.21/56.64
GPS-SELM (geo+rec)	35.10/51.68	38.03/54.05	39.40/55.01
GPS-SELM (geo+rec+int)	**37.09/54.52**	**39.96/56.14**	**41.35/58.10**

Table 2: Experiments on GPS-SELM on the area type. We report (top-1 accuracy/top-5 accuracy), averaged over ten evaluation trials.

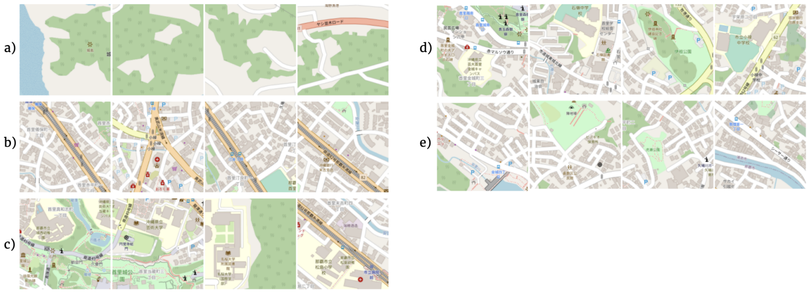

Figure 3: Linear interpolation in the latent space at equal intervals between representations of left and right images. We show 3 nearest neighbors in the latent space to each interpolated vector.

One possible reason is that area types exhibit strong correlations with their surrounding areas, which are captured through the reconstruction task.

4.4. Latent Space Interpolation

We present an in-depth analysis of the learned representations through a latent space interpolation experiment, illustrated in Figure 3. We utilize the GPS-SELM embeddings of a field tile and an urban maps, and perform a linear interpolation between the left and right maps. At each point along the interpolation, we search for the three nearest neighbors in the latent space and display the corresponding maps. This exploration of the semantically meaningful latent space reveals a gradual progression towards areas with shared features, particularly urban areas. Interestingly, it appears that some of the images exhibit a sense of continuity, resembling contiguous locations within the latent space.

4.5. Water Quality Prediction

Next, we apply GPS-SELM to predict *water quality indice* from the SustainBench dataset Yeh et al. (2021). Given a satellite image, the extracted features from the student backbone were

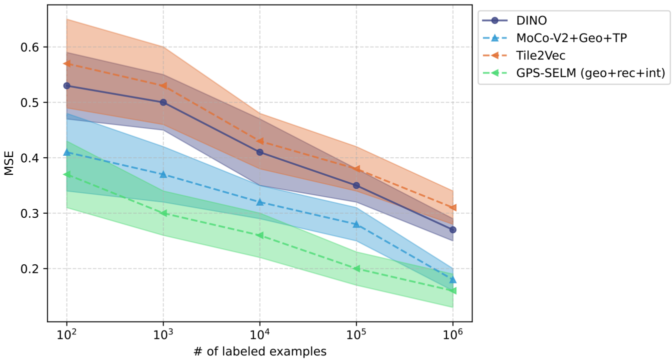

Figure 4: Experiments on GPS-SELF on estimating *water quality indice* from remote sensing images. We report results for different training set sizes, averaged over ten evaluation trials.

Pretrain	**NAIP (top-1 accuracy)**	**OSM (top-1 accuracy)**
Random Init	46.78	59.44
raw GPS	-	5.12
MUGS Zhou et al. (2022)	49.66(+4.72)	64.83(+7.70)
MoCo-V2 Chen et al. (2020)	51.65(+4.87)	66.87(+7.43)
DINO Caron et al. (2021)	51.30(+4.52)	65.99(+6.55)
MoCo-V2+Geo+TP Ayush et al. (2021)	53.70(+6.92)	66.10(+6.66)
Tile2Vec Jean et al. (2019)	51.89(+5.11)	68.34(+8.9)
GPS-SELM (geo)	54.33 (+7.55)	70.12(+10.68)
GPS-SELM (geo+rec)	53.99.(+7.21)	71.44(+12.0)
GPS-SELM (geo+rec+int)	**54.67** (+**7.89**)	**71.71**(+**12.27**)

Table 3: Land Cover Classification on NAIP dataset and OpenSreettMap (OSM) dataset. Results are averaged over 10 trials.

subsequently utilized for predicting water quality level Jean et al. (2016). In Figure 4, we present a comprehensive comparison of our method against the baseline models, including the method utilizing Tile2Vec embeddings Jean et al. (2019). The results clearly demonstrate the superior performance of GPS-SELM, showcasing its effectiveness in leveraging spatial representations and surpassing the existing state-of-the-art approaches. Notably, GPS-SELM demonstrates the ability to transfer knowledge from map images to remote sensing images. This is because map images and remote sensing images share common features such as vegetation and building structures. Furthermore, the utilization of maps can enhance generalization across tasks and regions, as maps tend to exhibit fewer noise and small fluctuations compared to remote sensing images.

4.6. Land Cover Classification

In addition, we perform experiments on two land cover classification tasks using 1) remote sensing images obtained by the USDA's National Agricultural Imagery Program (NAIP) DOI/USGS/EROS

Pretrain	AP_{50}
Randon Init	10.73
MoCo-V2 Chen et al. (2020)	15.42 (+4.69)
DINO Caron et al. (2021)	17.50 (+6.77)
MoCo-V2+Geo+TP Ayush et al. (2021)	17.72 (+6.99)
Tile2Vec Jean et al. (2019)	15.67 (+4.94)
GPS-SELM (geo)	18.04 (+7.31)
GPS-SELM (geo+rec)	17.99 (+7.26)
GPS-SELM (geo+rec+int)	**18.79** (+**8.06**)

Table 4: Object detection results on the xView dataset.

(2021), and 2) map images from OpenStreetMap (OSM). As done in Ayush et al. (2021), we use the images from the California's Central Valley for the year of 2016. The dataset consists of 100,000 training and 50,000 test images. In detail, we perform transfer learning experiments on land cover classification across 66 land cover classes. For the OpenStreetMap dataset, we randomly selected 100,000 training and 50,000 test images, and used 13 land type classes. Table 3 depicts the results for both remote sensing and OSM datasets. On the NAIP dataset, our method outperforms the randomly initialized weights by 8.89% and MoCo-V2+Geo+TP by 0.97%. On the OSM dataset, our method outperforms the randomly initialized weights by 12.27% and MoCo-V2+Geo+TP by 5.34%. The results can be explained by several factors. 1) Our approach incorporates prior knowledge about the spatial relationships between different land cover types, which is not explicitly captured by randomly initialized weights or standard self-supervised approaches. 2) GPS-SELM uses a multi-task learning framework that jointly optimizes geo-predictive classification tasks and intermediate clustering, improving generalization to new datasets.

4.7. Object Detection

We now report results for an object detection task. For object detection, we use the xView dataset Lam et al. (2018) consisting of high-resolution satellite images. The xView dataset contains 846 images of size 2000×2000 pixels. They are satellite images with bounding box annotations for 60 different class categories including passenger, airplane vehicle, etc. The dataset was divided into a training set of 700 images and a test set of 146 images. As done in Ayush et al. (2021), we process the images to create 416×416 pixels images by randomly sampling the bounding box coordinates of the small image and we repeat this process 100 times for each large image. Table 4 shows the object detection performance on the xView test set. Interestingly, the proposed method achieved the best results. Even though GPS-SELM was pretrained on map images, our approach achieves strong performance, highlighting the value of leveraging map-based architectures including on remote sensing downstream tasks. A potential factor contributing to this achievement is the model's capacity to discern objects through the incorporation of auxiliary tasks during pretraining, enabling a more comprehensive understanding of objects (e.g., POIs) and spatial relationships.

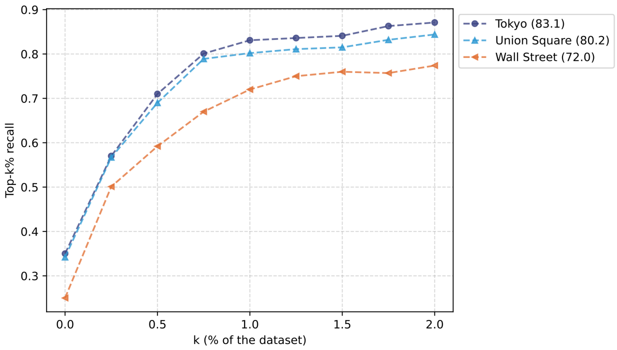

Figure 5: Top k% recall results obtained by utilizing the learned embedded space to retrieve map tiles based on given location images, where k% is the fraction of the dataset size. Results for three subsets: Tokyo, Union Square, and Wall Street. Top 1% recall values are shown in brackets.

4.8. From Embeddings to GPS Coordinates

We now assess the ability of GPS-SELM to link learned embeddings with appropriate GPS coordinates. We generated testing subsets from areas in Tokyo, Union Square, and Wall Street, each containing 5,000 locations. We investigate the recall performance when using the embedded space to retrieve corresponding map tiles given location images — how likely is the corresponding map image to be the closest within the space. Top-k% recall plots are shown in Figure 5, where top-k% recall is the fraction of cases in which the ground truth tile is within the top k% of best estimates. Remarkably, GPS-SELM exhibits strong performance, achieving a top-1% recall of over 72%. One can observe that Wall Street presents a more challenging scenario due to its distinct characteristics, including motorways and tunnels, which differ from the training set. Nevertheless, even under these conditions, our model is able to assign a high rank to corresponding map tiles.

5. Conclusion

We proposed a novel map-based approach to GPS coordinate embedding using self-supervised learning. The method leverages a student-teacher architecture to encourage the student network to learn meaningful embeddings that capture geographical and topographical features. We further introduced auxiliary tasks, including geo-predictive tasks, a high-level reconstruction task and an intermediate clustering task. The present algorithm outperformed several baseline methods in terms of the quality of the learned embeddings and demonstrated the usefulness of our embeddings on downstream tasks, such as predicting land price or land cover. Experimental results demonstrate that this map-based approach enables the model to capture more complex relationships between locations than previous baselines, by incorporating contextual information from surrounding locations. Hence, the proposed method has the potential to improve the performance of various location-based applications, such as clustering, and prediction.

References

Woven city. https://www.woven-city.global/, 2023.

Kumar Ayush, Burak Uzkent, Chenlin Meng, Kumar Tanmay, Marshall Burke, David Lobell, and Stefano Ermon. Geography-aware self-supervised learning. In *International Conference on Computer Vision*, pages 10181–10190, 2021.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *International Conference on Computer Vision*, pages 9650–9660, 2021.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Conference on Computer vision and Pattern Recognition*, pages 15750–15758, 2021.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

Sina Dabiri, Nikola Marković, Kevin Heaslip, and Chandan K Reddy. A deep convolutional neural network based approach for vehicle classification using large-scale gps trajectory data. *Transportation Research*, 116:102644, 2020.

DOI/USGS/EROS. National Agriculture Imagery Program (NAIP) Dataset. https://catalog.data.gov/dataset/national-agriculture-imagery-program-naip, 2021.

Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Learning representations by predicting bags of visual words. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6928–6938, 2020.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33:21271–21284, 2020.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

Juan C Herrera, Daniel B Work, Ryan Herring, Xuegang Jeff Ban, Quinn Jacobson, and Alexandre M Bayen. Evaluation of traffic data obtained via gps-enabled mobile phones: The mobile century field experiment. *Emerging Technologies*, 18(4):568–583, 2010.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Neal Jean, Marshall Burke, Michael Xie, W Matthew Davis, David B Lobell, and Stefano Ermon. Combining satellite imagery and machine learning to predict poverty. *Science*, 353(6301): 790–794, 2016.

Neal Jean, Sherrie Wang, Anshul Samar, George Azzari, David Lobell, and Stefano Ermon. Tile2vec: Unsupervised representation learning for spatially distributed data. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):3967–3974, Jul. 2019.

D Joshi. Inferring generic activities and events from image content and bags of geo-tags. In *ACM International Conference on Image and Video Retrieval*, 2008.

Darius Lam, Richard Kuzma, Kevin McGee, Samuel Dooley, Michael Laielli, Matthew Klaric, Yaroslav Bulatov, and Brendan McCord. xview: Objects in context in overhead imagery. *arXiv preprint arXiv:1802.07856*, 2018.

Ziwei Li, Ke Xu, Haiyang Wang, Yi Zhao, Xiaoliang Wang, and Meng Shen. Machine-learning-based positioning: A survey and future directions. *IEEE Network*, 33(3):96–101, 2019.

Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. 2017.

Xiaoqiang Lu, Xiangtao Zheng, and Yuan Yuan. Remote sensing scene classification by unsupervised representation learning. *Geoscience and Remote Sensing*, 55(9):5148–5157, 2017.

MLIT. Ministry of Land, Infrastructure, Transport and Tourism Dataset. https://nlftp.mlit.go.jp/ksj/gml/datalist/KsjTmplt-L01-v3_1.html, 2023.

Lichao Mou, Pedram Ghamisi, and Xiao Xiang Zhu. Unsupervised spectral–spatial feature learning via deep residual conv–deconv network for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 56(1):391–406, 2017.

OpenStreetMap contributors. OpenStreetMap . https://www.openstreetmap.org, 2017.

Jonathan Raper, Georg Gartner, Hassan Karimi, and Chris Rizos. Applications of location–based services: a selected review. *Journal of Location Based Services*, 1(2):89–111, 2007.

Noe Samano, Mengjie Zhou, and Andrew Calway. You are here: Geolocation by embedding maps and images. In *European Computer Vision*, pages 502–518. Springer, 2020.

Kevin Tang, Manohar Paluri, Li Fei-Fei, Rob Fergus, and Lubomir Bourdev. Improving image classification with location context. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1008–1016, 2015.

Chenyu Tian, Yuchun Zhang, Zefeng Weng, Xiusen Gu, and Wai Kin Victor Chan. Learning large-scale location embedding from human mobility trajectories with graphs. *arXiv preprint arXiv:2103.00483*, 2021.

Chen Wang. *Location based services and location based behavior in a smart city*. PhD thesis, Université de Lyon, 2016.

Christopher Yeh, Chenlin Meng, Sherrie Wang, Anne Driscoll, Erik Rozi, Patrick Liu, Jihyeon Lee, Marshall Burke, David Lobell, and Stefano Ermon. Sustainbench: Benchmarks for monitoring the sustainable development goals with machine learning. In *Neurips Conference*, 12 2021.

Yifang Yin, Zhenguang Liu, Ying Zhang, Sheng Wang, Rajiv Ratn Shah, and Roger Zimmermann. Gps2vec: Towards generating worldwide gps embeddings. In *International Conference on Advances in Geographic Information Systems*, pages 416–419, 2019.

Pan Zhou, Yichen Zhou, Chenyang Si, Weihao Yu, Teck Khim Ng, and Shuicheng Yan. Mugs: A multi-granular self-supervised learning framework. *arXiv preprint arXiv:2203.14415*, 2022.