RING-Net: Road Inference from GPS Trajectories using a Deep Segmentation Network

Emre Eftelioglu
efteli@amazon.com
Amazon
Bellevue, Washington, USA

Ravi Garg
ravigarg@amazon.com
Amazon
Bellevue, Washington, USA

Vaibhav Kango
kangov@amazon.com
Amazon
Bellevue, Washington, USA

Chintan Gohil
chintang@amazon.com
Amazon
Bellevue, Washington, USA

Amber Roy Chowdhury
amberch@amazon.com
Amazon
Bellevue, Washington, USA

ABSTRACT

Accurate and rich representation of roads in a map is critical for safe and efficient navigation experience. Often, open source road data is incomplete and manually adding roads is labor intensive and consequently expensive. In this paper, we propose **RING-Net**, an approach for **R**oad **IN**ference from **G**PS trajectories using a deep image segmentation **Net**work. Previous work on road inference is either focused on satellite images or GPS trajectories, but they are not compatible with each other when there is a lack of high quality data from either of the source types. Even though it is primarily focused on using GPS trajectories as its input, RING-Net architecture is flexible enough to be used with multiple data sources with minimal effort. More specifically, RING-Net converts raw GPS trajectories into multi-band raster images with trip related features, and infers roads with high precision. Experiments on public data show that Ring-Net can be used to improve the completeness of a road network. Our approach is promising to bring us one step closer to fully automated map updates.

CCS CONCEPTS

• **Computing methodologies → Machine learning**; **Supervised learning**; **Image segmentation**; • **Applied computing → Transportation**; • **Information systems → Geographic information systems**; **Global positioning systems**.

KEYWORDS

GPS Trajectories, Road Inference, Deep Learning, Segmentation

ACM Reference Format:
Emre Eftelioglu, Ravi Garg, Vaibhav Kango, Chintan Gohil, and Amber Roy Chowdhury. 2022. RING-Net: Road Inference from GPS Trajectories using a Deep Segmentation Network. In *The 10th ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data (BigSpatial '22) (BigSpatial '22), November 1, 2022, Seattle, WA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3557917.3567617

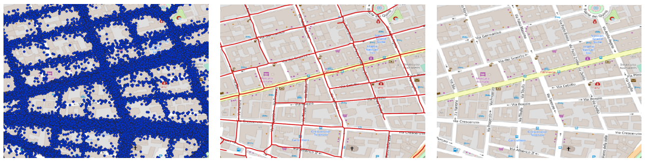

Figure 1: Illustration of the proposed approach. Given a set of GPS points in (a), the output is shown in (b). The corresponding OSM basemap is shown in (c).

1 INTRODUCTION

Accurate and rich representation of roads in a map is critical for safe, efficient, and user-friendly navigation experience. A complete map can help in a near optimal route planning for simple navigation from point A to point B, or in more complicated scenarios of multi-stop visits. Accurate maps enable not just reduction in navigation cost but also bring positive environmental impact through reduced fuel consumption due to efficient routing. Almost all applications available for road navigation usually source map data by either (i) buying a commercially available road map, or (ii) using crowd-sourced services, e.g., Open Street Map [30].

Commercial maps use a variety of approaches including traditional field surveys, customer feedback, and transportation officials' road updates to improve completeness [37]. Such efforts to manually improve maps are time consuming and expensive. Crowd-sourced road maps (e.g., Open Street Map (OSM) [30]), on the other hand, bring other challenges [15]. First, such data are not homogeneous - the quality and coverage depends on the location and the number of contributing volunteers [27]. Consequently, less populated areas (with less number of volunteers) have sparse information, making it difficult to validate the quality and completeness. Second, although some counter-measures are taken by the community, there is always a risk of vandalism, which may cause serious consequences [43]. For example, intentionally classifying a one-way road segment as two-way will pose a risk to navigation safety. Third, crowd-sourced maps may not be regularly updated everywhere [24]. Contributors such as the companies using this data as their main source may

BigSpatial '22, November 1, 2022, Seattle, WA, USA

ACM ISBN 978-1-4503-9531-1/22/11…$15.00
https://doi.org/10.1145/3557917.3567617

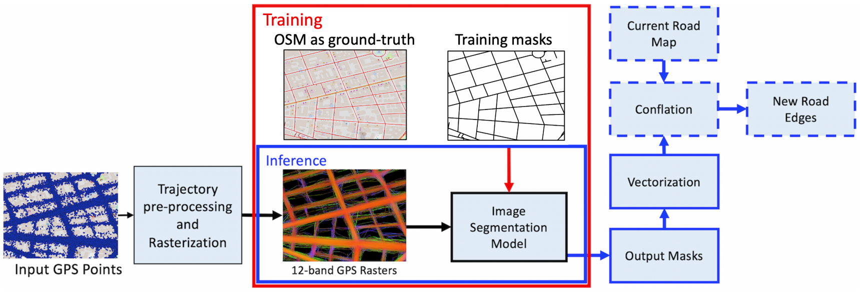

Figure 2: A potential end to end system architecture. Training inputs are depicted with red and inference in blue. Conflation steps are represented with dashed blue lines which can improve road network updates.

update their regions of interest more frequently. However, volunteers are not obligated to update maps at a regular cadence. Data from well-crowded locations may be more up-to-date, but data from remote regions may be stale due to less number of contributors or infrequent updates. To ensure efficiency and reliability, crowd-sourced road maps should be regularly updated. However, manual updates are labor intensive, time-consuming, and expensive. This becomes more evident for regions where pre-existing maps are very sparse, such as in the developing world [7]. Considering the ever changing road networks, automated map updates are desirable.

1.1 Problem Definition

To address these issues and make crowd-sourced road map updates automated, we propose a **R**oad **IN**ference approach from **G**PS trajectories using a deep image segmentation **Net**work, i.e., **RING-Net**. Our proposed approach is an image segmentation model that is being used with GPS trajectories (satellite image inputs are out-of-the-scope). However, the model has a flexible architecture which can be used for both GPS trajectories and satellite images, with minimal effort. Therefore, in case of having multiple data sources, these can complement each other when a specific source is lacking in a region.

RING-Net can be defined as follows: Given a set of GPS trajectories with their timestamps, RING-Net generates images with 12-bands that incorporate multiple trip related information such as traversal count, speed, acceleration, bearing change and cardinal directions. These images are then passed as inputs to a deep neural network based image segmentation model. Output from the model are skeletonized as images, followed by vector conversion to obtain output road edges. We use ground-truth road network from OSM to train RING-Net segmentation model.

Figure 1 illustrates an input set of GPS points and the corresponding predicted road of the proposed approach. On the right, the basemap from OSM [30] corresponding to the region is shown.

It is worth noting that in a typical use-case, there is usually a partially complete road network available in a region. Therefore, instead of re-discovering the whole road network, a conflation step [13, 45], which extracts the missing road segments with respect to a given map, can be used. However, in this paper, we focus our efforts to describe the road extraction and omit the conflation steps. We believe that using RING-Net together with a conflation approach can bring us one step closer to fully automated map updates without human intervention. In Figure 2, a potential end-to-end workflow is illustrated. The boxes in solid lines (red for training and blue for inference) are described in this paper. The boxes with dashed lines represent conflation to extract only the new roads in the inferred road network graph, as compared with the base map.

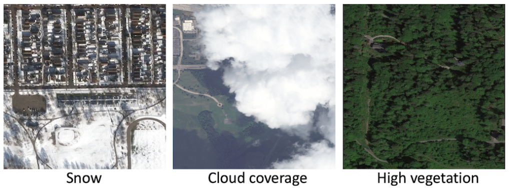

Figure 3: Potential problems associated with satellite images.

1.2 Related Work

Road extraction problem has attracted significant attention from researchers over the last decade. Related work in the broader road extraction problem can be split into two main research tracks based on the types of inputs used: a) road extraction from satellite images, and b) from GPS trajectories. Since these two main tracks have their own challenges and benefits, we will discuss these in the next sections.

1.2.1 Road Extraction from Satellite Images: Road extraction from satellite imagery is an active area of research [3, 23, 25]. These approaches can be broadly divided into three categories – (a) borderline extraction, (b) image (semantic) segmentation, and (c) graph extraction approaches. With the advancement in deep learning architectures, semantic segmentation approaches using neural network architectures such as UNet [33], Fully Convolutional Networks (FCN), and ResNet [17] have demonstrated state-of-the-art results in recent years [26] as compared to borderline extraction approaches using feature engineering and edge detection methods. A state-of-the-art model, **D-LinkNet** [47], extends LinkNet [10] architecture, an encoder-decoder style architecture based on ResNet [17], by adding a dilated convolution block between the encoder and decoder blocks. The dilated convolutions progressively increase receptive field to absorb larger context besides using kernels designed to extract long edge based road features. This solution achieved top IoU score of 0.64 on DeepGlobe'18 dataset [12].

However, inferring roads from satellite images at scale is still a challenging problem. Most research are conducted over limited public data over smaller regions, and use satellite images with predefined specifications (e.g., resolution, cloud cover, occlusions). In real world scenario, when images are collected, some roads may be occluded by tree/cloud-cover, off-nadir angle, and sun elevation based on the time of day. Figure 3 illustrates some of these problems associated with satellite images. Moreover, in most commercial map applications, procuring satellite image is costly, and not updated frequently by the vendors. Finally, for large regions, same resolution imagery may not be available or may be collected at different dates that cause complex model training and maintenance needs. Therefore, additional approaches using complementary datasets (e.g. GPS trajecories) are needed.

1.2.2 Road Extraction from GPS Trajectories: Road network inference from GPS trajectories are usually developed as standalone approaches (without considering any compatibility needs). These can be classified into 3 main categories: (a) Clustering based methods [11, 14, 20], (b) Merging (stitching) based methods [18, 19], and (c) Image based methods [6, 41].

Clustering based methods rely on grouping GPS points/lines to clusters based on distance metrics such as Euclidean, Manhattan, or Fréchet distance. Then each cluster is stitched with a rule defined on features such as linearity, curvature, etc. The stitched output is then converted to a graph to obtain the road network.

Merging based methods partition GPS trajectories into small segments, find similar ones by using speed, direction and distance between them, and add them to an initially empty map. Most work in this category use greedy approaches. First, GPS trajectories are segmented and added to the map if there aren't any similar edges in the graph. If there is already an edge around the same region, then the confidence level of that edge is incremented. Finally, given a threshold, the edges not passing a minimum configurable threshold are removed from the graph and the output is returned.

Image methods focus on using rasters created from GPS. First, GPS trajectories are rasterized (either as points or lines), several image processing techniques such as morphological erosion, dilation, etc. with spatial constraints are applied, and finally skeleton of the road network is extracted. The directionality and other information is often extracted by a topology learning step, where the same input GPS trajectories are map-matched with the output skeleton to provide connectivity as well as directionality. These methods perform well for relatively clean or less noisy GPS data. However, in a real world scenario with noisy and heterogeneous data across a large area with different densities, their performance often degrades. Moreover, by not extracting the relevant information such as speed, acceleration, and bearing changes (i.e., turns), these perform poorly when data has multiple trip modes, i.e., driving, walking, etc.

1.3 Challenges

Road extraction methods using satellite imagery may suffer from data related issues such as vegetation, sun elevation at the time of image capture, etc. Moreover, tuning any road extraction method to work across such varying conditions requires a substantial amount of imagery, which is often cost-prohibitive. On the other hand, using GPS data for the same brings its own challenges such as GPS noise and heterogeneity across a region. Finally, due to varying data availability among different sources, there is a need for flexible/compatible architectures.

Apart from the input data quality, annotated ground-truth data at large scale is not readily available for training. Using human annotation for millions of road segments is neither scalable nor efficient. Therefore, in this work, we propose to use road segments from OSM [30] as ground-truth for model training which eliminates the need for manual human annotations.

Finally, a reliable metric is needed to measure the performance with respect to topological correctness of predictions. Intersection-over-Union (**IoU**) (i.e. Jaccard Index), a widely used metric to measure pixel-wise accuracy, is not ideal for routing needs. For example, broken segmented roads may still exhibit a high IoU value, whereas a low IoU value may occur due to slight displacement from the ground-truth, but matter little for routing needs.

1.4 Contributions

RING-Net is a GPS trajectory focused approach to infer roads. However, its architecture is based on a satellite image based method, i.e. DLinkNet [47], with some improvements, making it compatible with both data sources. In this RING-Net makes the following contributions:

- We propose a novel approach to rasterize GPS trajectories, which aggregates multiple trip related features in a single image.
- We propose a scalable ground-truth road mask extraction approach for training, using OSM provided inputs and image processing algorithms.
- We use GPS data as input by modifying the state-of-the-art DLinkNet [47] architecture. We improve the topological connectivity of the output by embedding spatial attention mechanism [16] and centerline based loss function [36] that calculates loss over image skeletons instead of conventional mask outputs.
- We use a topological aware metric [5] and propose a novel variant, I-Topo, for measuring road extraction quality at intersections.

1.5 Scope and Outline

1.5.1 Scope: This paper proposes RING-Net, a road extraction model from GPS trajectories. Despite the large volume of research on road extraction from satellite images, we omit the comparisons with these approaches. Yet, it should be noted that our proposed architecture is very similar to another satellite image based method, i.e. DLinkNet [47], making it highly compatible with any satellite image dataset (if trained with). Previous approaches on road extraction from GPS trajectories do not usually consider large regions for their experiments (either due to the lack of available open-source GPS trajectory datasets or due to the scalability limitations). Thus, those are not-comparable with the proposed approach. However, to provide readers with more insights about the output quality, we show examples with most common datasets (i.e., Geolife [46], T-Drive [44]), and conduct more comprehensive experiments on Rome Taxi Trajectory dataset [8].

Model training is done on a proprietary GPS trajectory dataset collected for a month from a 100 square-mile region. We describe data preparation and training details in the following sections, but actual examples of the training data are omitted.

Finally, there is usually a high amount of noise associated with the data from GPS sensors. Therefore, our method performs better with increased accumulation of GPS traversals over a road edge (e.g. noise is randomly distributed spatially around the actual position of a GPS receiver). In our experiments, we use a dataset that was collected for 1 month (February 2014). We anticipate the precision of the outputs to improve with larger timeframe.

Finally, we can consider RING-Net as a complementary approach to improve satellite image based solutions for smaller datasets and a standalone solution for larger ones. This is also facilitated by using a similar architecture as the DLinkNet [47] segmentation architecture.

1.5.2 Outline: The rest of the paper is organized as follows: Section 2 describes the preliminaries and the model architecture. Section 3 starts with how the GPS trajectory datasets are prepared to be used in a segmentation framework followed by the training details. Section 4 describes the evaluation metrics and shows example outputs from both qualitative and quantitative perspective. Finally, Section 5 summarizes the contributions and previews future work.

2 PRELIMINARIES

In this section, we define some basic terms used in this paper, followed by the neural network model architecture.

2.1 Definitions

2.1.1 GPS Trajectory: A trajectory is a set of temporally ordered points $p_{1,2,...,m}$, where each $p_i \in N$ represents a location (Lon, Lat) on the surface of Earth and a recording timestamp t_i. Thus, a trajectory $tr_i \in TR$ is a representation of continuous movement with a set of discrete points.

2.1.2 Spatial Network (Road Map). is a set of nodes and edges, where each node, $n \in N$, is associated with coordinates over the surface of Earth. A road network graph is represented in the form of a spatial network. Usually, nodes represent the intersections of streets in the road network. Edges, E, are the roads connecting any two nodes. Thus, a spatial network graph, $G = (N, E)$, is a subset of a complete graph G_c, and the number of edges $|E| << (n\times(n-1))/2$. Since a typical road network has one and two way roads, it is a directed graph and not planar (e.g., overhead bridges breaks the planarity). However, for simplicity of analysis, this work considers spatial network as a planar and undirected graph.

2.1.3 Road Inference. Given input data sources such as satellite images and GPS trajectories, road inference aims to infer spatial network consisting of nodes and edges in such a way that the inferred graph represents a routable map. In this paper, road inference is focused on identifying road segments traversed by motor vehicles.

2.1.4 Semantic Segmentation. is the task of classifying each pixel of an input image into a finite and pre-defined categories. Semantic segmentation has been used to address many applications in geospatial domain such as land cover classification, building outline detection, and road extraction from satellite images. For road extraction problem, semantic segmentation based approaches typically classify each pixel into a foreground class (belonging to road) and a background class (non-road features in the image).

2.1.5 Rasterization. is the process of creating a grid representation of vectors in a bounded area. Formally, given a finite area A and a grid cell size c, the raster space R can be defined as the set of grid cells C overlapping with A and $|C| = |A/(c^2)|$. Rasterization represents a vector V (e.g., point, line, polygon, etc.) in R such that each grid cell intersected by V is in $V_R \subset C$.

2.2 Model Architecture

RING-Net adopts a semantic segmentation based D-LinkNet architecture [39], which has demonstrated state-of-the-art results in road segmentation literature on open source datasets. The original D-LinkNet architecture is based on an encoder-decoder architecture to extract the road segmentation mask from satellite images. In RING-Net, we modify the architecture to accept any number of input channels. This modification allows RING-Net to improve learning by including different GPS rasterization methods. We further modify the network architecture by introducing a spatial self-attention mechanism to help network learn global context in the inputs.

2.2.1 Spatial Self-Attention: This module captures spatial interdependencies between any two positions in the input feature maps. Similar to work in [16], the feature at each location is updated by aggregating the features obtained by weighting the input features, where weights are determined by the similarity of the input features with the feature at the given location. The intuition behind using such mechanism for the road extraction problem is that the driving behavior (inferred from GPS trajectories) is usually consistent across different regions, and attention-aware learning can help boost performance by extracting correlated features at different locations. We add the spatial self-attention layer between the third encoder and the last encoder in the down-sampling part of the DLinkNet architecture. Spatial self-attention layer receives a smaller input feature map as compared to the previous layers resulting in a smaller number of extra learning variables. In our experiments, this configuration provides best results instead of self attention layer at other encoder locations. The network architecture used in RING-Net is shown in Figure 4.

2.2.2 Centerline Dice Loss Function: Traditional loss functions (e.g., Dice Loss [38]) used in semantic segmentation based methods usually focus on the volumetric accuracy of predictions. Dice loss function is biased on correctly segmenting large clusters of areas in an image as belonging to different classes. However, for road extraction frameworks, the area of the region of interest is usually smaller compared to the other parts of the image. Moreover, the connectivity of output segments is more important than their overlapping area since we are not interested in detecting the width of a road, but its linearity and connectivity. Recently, a centerline based Dice function was proposed for segmentation of vascular structures in medical images [31, 36]. In RING-Net, we use a weighted average of the traditional Dice loss and Centerline Dice Loss to ensure both the topology and connectivity in the output masks. The centerline

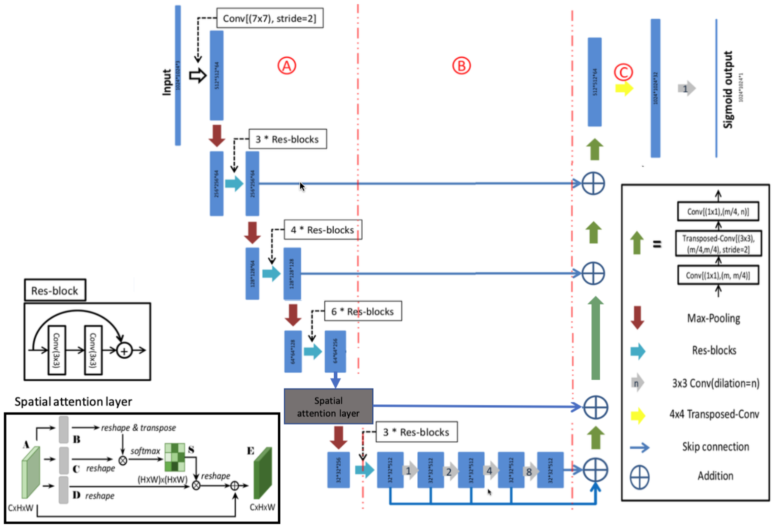

Figure 4: D-LinkNet architecture with spatial self-attention. Blue rectangles represent convolutional filters. Part A and Part B are decoder and encoder of LinkNet respectively. Part B has layers of dilated convolution to increase receptive field.

dice loss function can be mathematically represented as following:

$$L_c = (1 - \alpha) \times (1 - softDice) + \alpha \times (1 - softClDice)\,,$$

where $softDice = 1 - \frac{\frac{2 \cdot a}{b}}{\frac{a+b}{b}} = \frac{2 \cdot \frac{a}{b}}{\frac{a}{b}+1} = \frac{2 \cdot IoU}{IoU+1}$ (a here is defined as true positive and true negative pixel classes and b is true positive + false positive + true negative pixel classes). For $softClDice$ function, a and b are measured on finite pixel width skeleton representation of the road network. In our experiments, we observed that the value of α affects the result quality substantially, and the best results are obtained for $\alpha = 0.4$. For larger α values, we observed that the GPS noise is starting to get incorporated in the output masks, whereas for smaller α values output was not properly connected. Hence, we use $\alpha = 0.4$ for our experiments results described in this paper.

3 PROPOSED APPROACH

RING-Net uses rasterized GPS data in a semantic segmentation framework. Given a set of GPS trajectories, we perform a data cleaning to filter out inconsistent GPS points (e.g., from random and large jumps in location). We then create the GPS rasters and use them as input to a semantic segmentation based neural network to infer road network. When devising this approach, we consider several challenges. First, GPS data can have multiple travel modes such as walking, driving, etc., making the problem harder. Second, a binary trajectory *existence* raster may not be enough (i.e., traversed vs. not-traversed) to capture the trip details. Third, a single model may not be enough for completeness (when there is no GPS data and when satellite images are stale or obscured). Thus, we need to create an architecture which can be used for different data sources, i.e., satellite images and GPS data.

Under these assumptions, we propose a data cleaning step to eliminate the non-driving data as well as the GPS noise. We also propose a GPS rasterization method which can be used in any multi-input channel semantic segmentation framework.

3.1 Preliminary GPS Data Cleaning

GPS data has inherent noise from various sources such as atmospheric changes, urban canyons, sensor quality, etc. Therefore, a GPS point can be represented by $p_t = p_{0t} + n_t$, where p_{0t} is the real location as time t and n_t is the noise. It is impossible to mitigate the effects of noise completely. However, a variety of approaches such as extended Kalman Filters [32], Spline Fitting [35], Line Simplifications [22], GPS Data Clustering [2], map-matching [29] were proposed.

In our approach, we consider three issues with these. First, most of the devices already apply Kalman filter to their sensor output. Thus, running a similar approach on already processed data will introduce more bias. Second, map-matching based approaches are not applicable when we try to infer the road network. Third, simplification based approaches introduces more uncertainty due to the usage of heuristics based parameters, but a complex enough model may already infer a true position. Therefore, when working with the GPS data, we remove only the points which we consider carry too much noise (by considering physical limitations of vehicles). However, we do not modify the data in any other way. In this simple heuristic, we use practical limits of various driving-based physical properties. Our GPS data cleaning approach makes use of three properties: a) speed, b) turn direction, and c) acceleration. Below we outline the details of those filters, but we do not specify their individual thresholds. Those can be selected by specified physical limitations.

3.1.1 Speed Based: For each GPS trajectory $tr_i \in TR$, consecutive pair of points $(p_t, p_{t-1}) \in P_{tr_i}$ can be considered as a line segment, where using the positional information (longitude, latitude) and time lag, driving properties such as "speed", and "acceleration" can be calculated. We assume that the data has mixed trip modalities (walking, driving, etc.) and thus we use a speed threshold to flag the points as "walking". In the future, we plan to use more sophisticated models, e.g., LSTM [28] at this stage.

3.1.2 Turn Based: Similarly, we observe that when a vehicle is moving slowly, the effect of noise becomes more pertinent. A slower vehicle will report more zig-zaggy position coordinates than a fast moving one. To eliminate these, we split trajectories into line segments of consecutive point pairs. Thus, a $tr_i \in TR$ is a set of $line_j \in tr_i$ where each $line_j = F_{line}(p_t, p_{t-1})$. We then calculate the compass heading (bearing) between each. Finally, points which has physically impossible turn angles are removed (shared point between two consecutive lines). For example, traveling with $60mph$, it is physically impossible for a car to turn 90 degrees in 5 seconds.

3.1.3 Acceleration Based: Finally, we check the physical possibility of an acceleration/deceleration. For example, an ordinary passenger car will not accelerate with a rate > 10 m/s^2. Using physical limitations of acceleration and deceleration (these physical-impossibility thresholds are out-of-the-scope of this paper), GPS points which shows physically impossible movement are removed.

3.2 GPS Data Rasterization

Conversion of GPS trajectories into raster images provides an opportunity to leverage the broader image processing research. In addition, using RING-Net in compliance with a satellite image based model (e.g., the approach in [47]), compatibility of data will reduce the need for a separate workflow creation. Finally, it provides an ensemble of all the GPS trajectories in a location by aggregating them over the grid cells (i.e., pixels).

Previous work on GPS rasterization considered the problem from a binary/grayscale image generation perspective. As described in the Definition 2.1.5, each vector intersecting a grid cell can be flagged as traversed, which creates a binary image that represents traversals at that pixel location. An improvement over a binary representation is to accumulate the traversals, resulting in a grayscale image. We take this one step further and develop a rasterization method incorporating the trip information in 12 individual bands. Thus, for $|C|$ cells in the GPS raster, each cell $c \in C$ stores information related to TR. These are the number of traversals (i.e., how many times a cell c is intersected by a trajectory), average speed, bearing change (to represent the turns), acceleration, and the percentage of 8 cardinal directions (N, NE, E, SE, S, SW, W, NW). By stacking multiple bands of different driving behaviors, a single GPS raster can be used to identify location popularity, speed footprint, turns, directions, etc., and also provides a more strong input feature to learn the required attributes.

GPS rasterization algorithm start by defining an empty raster space. We empirically selected $2m$ resolution for our experiments. To have contiguity in a trip, instead of working with GPS points, we create consecutive pairwise lines. Given a GPS trajectory with N points, there are $N-1$ pairwise lines. For each line $line_j \in tr_i$, we calculate the speed, acceleration, bearing change (compared to previous $line_{j-1}$) and the cardinal direction of the line. This process has a cost of $O(N)$ since by a single pass over the points in the trajectory, all these can be calculated. Finally, lines are converted to cell coordinates using Bresenham's line generation algorithm [9] with a slight change. Bresenham's line generation algorithm creates the pixel coordinates of a line from its start cell to the end cell. However, to prevent multiple writings to the same pixel for each consecutive line (one line's end coordinate is the other's start coordinate), we modify the algorithm to not write to the end cell. An example GPS raster image is shown in Figure 5. The example is generated from the same GPS points in Figure 1.

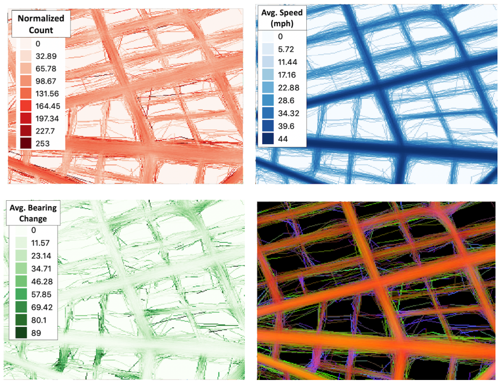

Figure 5: Example raster image created for the GPS points in Figure 1. Top-left depicts normalized counts of traversals per pixel, top-right represents average speed, bottom-left represents bearing change (notice the colors at intersections) and bottom-right depicts with all 12-bands.

3.2.1 Count Band Normalization: When GPS rasters are generated, the speed, acceleration and bearing change bands are created as averages. The cardinal directions are created as percentages. However, the count band is the count of the number of times a GPS line intersects with a grid cell. From a temporal perspective, as well as the non-homogeneous trip density across a region, the count band can vary between $0 - \infty$. To create a road inference model that can work across different timeframes, we need to normalize the counts to a specified range (i.e. 0-255). Several normalization techniques is possible such as log normalization, min/max normalization, etc. Since we want consistency and similar values even for different ranges (e.g. different locations, different timeframes, etc.), we use cumulative distribution function (CDF) based normalization. Thus, we create the histogram of counts across the area of interest, and calculate the count distributions and replace the count values with the percentage of pixels with less than the value. An example output of this process is shown in Figure 6. For the sake of comparison a corresponding binary raster is also added.

3.3 Model Training

3.3.1 Training Data Preparation: We use the 12-band GPS rasters described above as input to RING-Net. Since predicted output of or

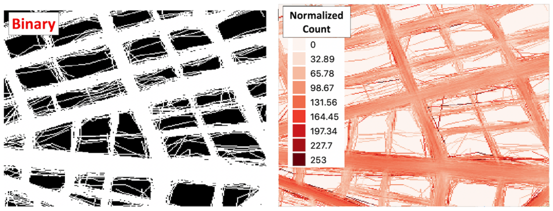

Figure 6: Comparison between binary and normalized count raster images.

model is a road mask, we also require the ground-truth labeled data with road and non-road annotations for the training set. In OSM data, road types are annotated by "Highway" field. We filter these by only driveable roads. Further, we clip the road line vectors from the dataset if they do not have any overlapping non-zero value GPS raster pixels. This is a critical step in our training data preparation, as using raw line vectors from OSM data may make the model infer roads from non-traversed locations in the GPS rasters. The road vectors after this cleaning process are rasterized as binary ground-truth mask images, with a configurable finite pixel width around each line vector. Figure 7 shows an example of a ground-truth mask generated out of the road network vectors of OSM.

We use an input raster size of 1024 × 1024 pixels to output a binary segmented mask of the same size. For that purpose, the 12-band GPS raster and corresponding ground-truth mask is subtiled into images of 1024 × 1024 pixels.

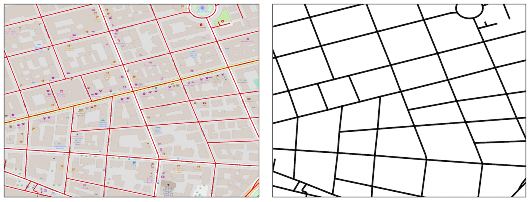

Figure 7: Illustration of ground-truth mask generation. Left shows the ground-truth vectors in red, and right shows the ground-truth mask generated for training.

3.3.2 Model Training Details: For training, we used a proprietary dataset consisting of 20000 GPS raster and ground-truth mask pairs collected from continental U.S. When creating the subtiles, we generated them with 100 pixel overlaps to eliminate the continuity issue at raster edges, which may prevent capturing information at the borders. Training parameters are selected as follows: we used the DLinkNet architecture with Self-Attention mechanism. When training, we selected the Adam optimizer [21]. The initial learning rate was $1e-3$ and used a step reduction in learning rate by decreasing it by factor of 5, if model does not show an improvement in the loss value for 10 successive epochs. Our loss function is a weighted loss function consisting of Dice loss and Centerline Dice Loss Function as described in Sec. 2.2.2. An interesting insight from the training is that the model learns quickly about most of the roads (as can been seen from the high accuracy values early), but continues to improve in later iterations. This is due to the fact that presence of GPS data is already a good indicator of presence of a road at the corresponding location.

4 EXPERIMENTAL EVALUATION

4.1 Evaluation Dataset

There are several open GPS datasets available. Among these, Geolife [46] and T-Drive [44] are the most widely used ones in the literature. While these are volume-rich, they have a low sampling rate (i.e., > 30 seconds) and collected from mixed trip modes (walk, bus, drive, flight) which leaves very small set of samples to experiment with once filtered. Yet, we still provide an example on the T-Drive dataset in Section 4.3.3.

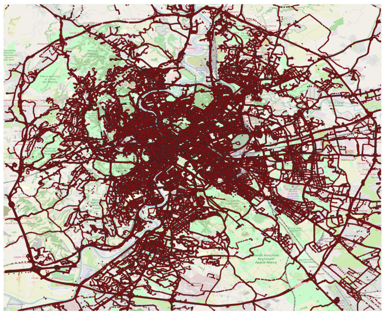

Figure 8: Visualization of GPS trajectory dataset used for experimental evaluation (GPS points in red). There are 320 participant taxis with 19 million GPS points.

To do a comprehensive analysis, we used a dataset with a higher sampling rate in a relatively busy area. One such dataset is collected from Rome, Italy in Feb.2014. In the study, ≈ 320 taxis collected trajectories for a month with a sampling rate of ≈ 7 seconds [8]. This data has ≈ 19 million GPS points, most of which are inside the circle highway around Rome. Figure 8 visualizes the GPS points in red from this dataset overlaid on the Open Street Map [30].

4.2 Evaluation Metric

Several methods have been introduced to evaluate the connectivity and geometric correctness of predicted road network. Some of the early work in this area use a connectivity metric of shortest path distances, but this is not reproducible due to random selection of road segments [42]. Spacenet Road Detection Challenge 2018 [40] introduced another novel metric called Average Path Length Similarity (**APLS**) based on the notion of betweenness centrality, which measures number of times a node appears on a shortest path. However, multiple shortest road segments may be present in an area with high road densities. Moreover, a path which is close to a shortest path would not receive a partial score. Therefore, this metric may not fully reflect quality of the extracted road network. Finally, DeepRoadMapper [23] focuses on connected roads by using a ratio (called connected road ratio) of road segments detected without discontinuities with respect to ground-truth segments.

In our work, we compare inferred road network with OSM data treated as ground-truth to measure accuracy of the predicted road geometry and topology. We adopt **Topo** [5] that measures topology and proximity of road network within a certain threshold (selected as 8 meters in our experiments) when compared to ground-truth network from OSM. This metric allows us to investigate the quality of the output by its topology and linear geometry, with familiar classification notions of precision and recall using the concept of holes and marbles as described below.

$$precision = \frac{matched\ marbles}{total\ marbles} \ and\ recall = \frac{matched\ holes}{total\ holes}$$

Topo, however, inadequately penalizes connectivity breaks to determine if inferred roads can be practically used for map building. We introduce a novel variant that intends to penalize deviations from correct topology more heavily even when the total matching road length is high. This metric works on road intersections and called as **I-Topo**. For I-Topo calculation, instead of generating regular interval holes over the road network, we generate holes at

Tile ID	Prec.	Rec.	F1
Rome-Tile-22	0.911	0.198	0.325
Rome-Tile-4	0.91	0.225	0.361
Rome-Tile-11	0.91	0.499	0.645
Rome-Tile-13	0.907	0.282	0.43
Rome-Tile-12	0.901	0.507	0.649
Rome-Tile-14	0.9	0.209	0.339
Rome-Tile-15	0.899	0.293	0.442
Rome-Tile-3	0.898	0.348	0.502
Rome-Tile-10	0.895	0.381	0.534
Rome-Tile-17	0.895	0.366	0.52

Table 1: Top-10(precision) topo results for Rome tiles.

intersections (or junctions, hence, I-Topo) only. When calculating the score, this increases the given importance to the intersections and penalizes common issue of connectivity breaks in inferred roads. A figure showing the key idea for Topo/I-Topo calculation is illustrated in Figure 9.

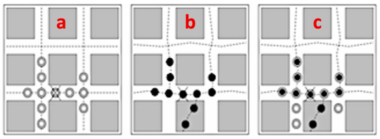

Figure 9: Illustration of Topo metric. a) holes (radius r) over OSM roads. b) marbles (interval of r) over predictions. c) holes and marbles are matched for evaluation.

4.3 Experiment Results

To provide a detailed analysis over the Rome GPS Trajectory dataset, we first split the region into 24 tiles. We then split each tile into subtiles, each of size 1024x1024 pixels to give as input to the model. Predicted output road masks from the model are stitched back to the original tile size. These stitched masks are smoothened to make them morphologically continuous, and to avoid issues associated with the skeletonization process. Smoothening is done by morphological closing with square elements of 5 pixels, followed by filtering all objects smaller than 100 pixels, and closing the holes in the mask. Post this, a graph is extracted from the image skeleton (SKNW [1]). Overlap region of about 100 pixels in tiles sharing a border aids connectivity of roads during skeleton creation. The road skeleton graph is geo-referenced to produce vector output for a larger region. Finally, these graphs are evaluated against the OSM road network for the same boundaries.

Table 1 and Table 2 show the Topo and I-Topo scores for each of these tiles. Note that the low recall values are due to incomplete coverage of the region in the GPS data. Interestingly, the recall values are higher for tiles with higher number of edges (total holes in the table is a linear function of the number of edges). This indicates that the busy areas which were traversed more by the taxis in the dataset have increased recall values. We expect that a larger accumulation and a higher coverage of GPS data over a region will help improve the recall values, with potentially improving precision as well. In summary, RING-Net achieves an average of 0.92 precision and 0.31 recall Topo scores for an 8 meter radius. For the same radius, I-Topo scores are 0.3 precision and 0.12 recall. Lower I-Topo scores are due to the GPS noise coupled with the sampling rate (7*s*), which cause the road intersections being predicted curvier than 90 degree angled lines. We anticipate that higher GPS sampling rates of $1-3$ seconds can improve geometry at intersections substantially.

Tile ID	Prec.	Rec.	F1
Rome-Tile-21	0.75	0.061	0.113
Rome-Tile-22	0.5	0.066	0.117
Rome-Tile-23	0.5	0.057	0.102
Rome-Tile-12	0.366	0.214	0.27
Rome-Tile-7	0.354	0.225	0.275
Rome-Tile-6	0.338	0.262	0.295
Rome-Tile-20	0.333	0.039	0.07
Rome-Tile-11	0.304	0.221	0.256
Rome-Tile-17	0.254	0.123	0.166
Rome-Tile-19	0.226	0.069	0.106

Table 2: Top-10(precision) i-topo results for Rome tiles.

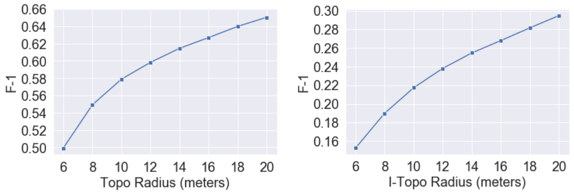

Figure 10: F-1 scores for different Topo and I-Topo radii.

4.3.1 *Effect of Selected Topo/I-Topo Radius:*

As a next step in our experiments, we measure the effect of increasing hold radius in Topo and I-Topo metrics from 6*m* to 20*m*. In the previous set of experiments, we selected the Topo/I-Topo radius as 8*m*. For parallel roads, to not over-score the model output, 8*m* is a valid distance to consider. However, depending on the location (such as crowded areas, urban canyons, roads with many lanes, etc.), GPS signal can be spread over a much higher radius and may cause model to predict slightly shifted roads. Such shift in predicted outputs will not give credit to the model output metrics for lower Topo radius. In Figure 10, the F-1 scores for both Topo and I-Topo are shown (precision and recall are omitted due to space constraints). As expected, scores are increasing for both metrics with increasing radii. Even though this trend is expected, we anticipate less impact for datasets that are collected from better GPS sensors.

4.3.2 *Visual Inspections:*

To visually validate the results, we select a tile with high precision value (Tile 11) to compare with its corresponding ground-truth road network vectors. We provide a zoomed view of two most prevalent road topologies in our datasets, i.e., complex intersections and grid roads. Figure 11 shows several examples of roads predicted at grid topologies. These outputs show that the grid topology is well captured by the RING-Net. However, the discontinuities on several roads shows the effect of lacking enough accumulation of GPS traversals. Finally, it is also worth noting that the output is not post-processed for topology refinement.

Figure 12 shows the output for complex intersections. Even though the output vectors have some artifacts (e.g., discontiguities and partially learnt lanes), especially on high width roads, it

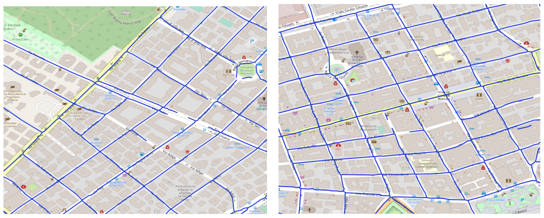

Figure 11: Output for grid structured roads. Even though the GPS data is sparser over more residential areas (less density of taxi traversals), the RING-Net recovered most road edges.

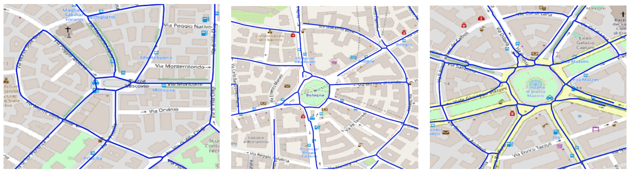

Figure 12: RingNet output for complex intersections.

still captures most of the roads correctly. These artifacts are not the result of the segmentation, but happen due to wider segmentation masks and their centerline skeletons, and can be further improved by tuning the skeletonization and vectorization parameters.

Even though we do not have the same input datasets, we also investigate the results from comparable research in [6] and [34]. However, since the given results are either from a dataset that was collected from a too small region or that is not openly available, we could not replicate the results. Thus, we compare our results with the reported metrics in these works. In [6], Biagioni et. al. reported the Topo precision of their approach as 0.7 for 7 months of GPS data with 8 m radius. The F-1 score is 0.65. In [34], the reported F-1 score is 0.45 for TaxiJN and 0.5 for TaxiBJ for a radius of 8 meters. Also, it is worth noting that both of these aforementioned methods are using a post-processing (e.g., topology refinement) and filtering steps ([34] using a filter of 10 traversals for TaxiBj and 5 traversals for TaxiJN), which helps having a significantly improved precision.

In summary, experiments demonstrate that RING-Net predict road edges with a high precision and a high recall when there is sufficient GPS data accumulation. We believe the output quality can be further improved by similar post-refinement approaches as used in [6] [34]. Other potential improvements in the future are discussed in Section 5.

4.3.3 Additional Experiments on Other Datasets: Since T-Drive [44] data is too sparse to do a comprehensive experimental analysis at a city scale and the sampling rate is too high to capture intersections well enough (e.g., >30 seconds), we decided to select a single tile which was previously experimented with other approaches in [34]. For the selected tile which is a 5km × 5km region, our model Topo precision is 0.72 and F-1 score is 0.43. A small example of a complex intersection in this tile is shown in Figure 13. On the top figure, only T-Drive dataset is used, bottom figure shows the output when T-Drive and Geolife [46] datasets are used as a single input. The improvements indicates that the additional data improves the model performance. From the metrics perspective, when the Geolife and the T-Drive datasets are used together, the precision is increased to 0.81, whereas F-1 score is increased to 0.51. Finally, it is worth noting that road predictions at the same region as the intersection shown in Figure 13 was illustrated with multiple approaches in [34]. Even without post-processing our approach provides better topology and Topo/I-Topo scores (Figures from [34] not provided in this paper due to copyright concerns).

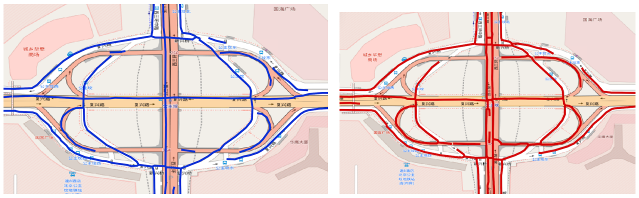

Figure 13: A complex intersection evaluated using two different approaches. Left shows output using T-Drive dataset, right shows output with better connectivity when both T-Drive and Geolife Datasets are used together. Due to sparseness and low sampling rate, some roads are split into small partitions. Outputs are not post-processed.

5 CONCLUSION AND FUTURE WORK

In this paper, we propose RING-Net, a road inference approach from GPS trajectories using an image segmentation framework. To that end, we used a novel rasterization method to encode GPS trajectories into a 12-band raster. This multi-band raster has rich information to learn road network using the ensemble of a set GPS trajectories. We use this raster image as an input to an image segmentation model, followed by vectorization to obtain output road vectors. Ring-Net improves the accuracy and the completeness of a road network to provide a safe, efficient, and user-friendly navigation experience. Our model can be used together with a satellite image based approach and reduce model maintenance needs in a production setting by consolidating different data sources in a single model architecture.

In future, we plan to improve the quality from multiple aspects. First, we plan to experiment with varying input GPS trajectory data sizes. However, the open source high frequency (i.e., $< 10s$) GPS trajectory datasets are not abundant. Therefore, we will investigate synthetic GPS trajectory generation methods to evaluate the model with different densities. Second, we plan to improve the connectivity of the output by using post-refinement techniques. In our experiments, we observed occasional discontinuities in the output. Thus, there is a need for a post-processing to eliminate these. For example, [34] uses map-matching to improve the topologies, whereas [4] applies a neural network based method. We also plan to synergize our approach with graph inference methods such as [18] to further improve the precision, and avoid the need of vectorization from masks. Furthermore, we also plan to investigate methods that can further leverage the GPS raster images created in our approach. For example, the bands representing the cardinal direction of traversals can be used for predicting the road directionalities. Similarly, speed limit can be predicted using the speed band and road class (e.g., residential, highway, etc.) can be predicted since it is typically a function of width and speed at a given road.

REFERENCES

[1] 2019. Library on network building from skeleton images. https://github.com/Image-Py/sknw.

[2] Farhan Ahammed, Javid Taheri, and Albert Zomaya. 2011. LICA: robust localization using cluster analysis to improve GPS coordinates. In *Proceedings of the first ACM international symposium on Design and analysis of intelligent vehicular networks and applications*. 39–46.

[3] Favyen Bastani, Songtao He, Sofiane Abbar, Mohammad Alizadeh, Hari Balakrishnan, Sanjay Chawla, Sam Madden, and David DeWitt. 2018. Roadtracer: Automatic extraction of road networks from aerial images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4720–4728.

[4] Anil Batra, Suriya Singh, Guan Pang, Saikat Basu, CV Jawahar, and Manohar Paluri. 2019. Improved road connectivity by joint learning of orientation and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10385–10393.

[5] James Biagioni and Jakob Eriksson. 2012. Inferring road maps from GPS traces: Survey and comparative evaluation. In *In Transportation Research Board, 91st Annual*.

[6] James Biagioni and Jakob Eriksson. 2012. Map inference in the face of noise and disparity. In *Proceedings of the 20th ACM SIGSPATIAL Conference*. 79–88.

[7] Derrick Bonafilia, James Gill, Saikat Basu, and David Yang. 2019. Building High Resolution Maps for Humanitarian Aid and Development with Weakly-and Semi-Supervised Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. 1–9.

[8] Lorenzo Bracciale, Marco Bonola, Pierpaolo Loreti, Giuseppe Bianchi, Raul Amici, and Antonello Rabuffi. 2014. CRAWDAD dataset roma/taxi (v. 2014-07-17). *CRAWDAD wireless network data archive* (2014).

[9] Jack E Bresenham. 1965. Algorithm for computer control of a digital plotter. *IBM Systems journal* 4, 1 (1965), 25–30.

[10] Abhishek Chaurasia and Eugenio Culurciello. 2017. LinkNet: Exploiting encoder representations for efficient semantic segmentation. *2017 IEEE Visual Communications and Image Processing (VCIP)* (Dec 2017). https://doi.org/10.1109/vcip.2017.8305148

[11] Jonathan J Davies, Alastair R Beresford, and Andy Hopper. 2006. Scalable, distributed, real-time map generation. *IEEE Pervasive Computing* 5, 4 (2006), 47–54.

[12] Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raska. 2018. Deepglobe 2018: A challenge to parse the earth through satellite images. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 172–17209.

[13] Thomas Devogele. 1996. Integration and matching of geographic databases. In *Building a Multi-Scale Database with Scale–Transition Relationships*. 337–351.

[14] Stefan Edelkamp and Stefan Schrödl. 2003. Route planning and map inference with global positioning traces. In *Computer science in perspective*. Springer, 128–151.

[15] Andrew J Flanagin and Miriam J Metzger. 2008. The credibility of volunteered geographic information. *GeoJournal* 72, 3-4 (2008), 137–148.

[16] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu. 2019. Dual Attention Network for Scene Segmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3146–3154.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778.

[18] Songtao He, Favyen Bastani, Sofiane Abbar, Mohammad Alizadeh, Hari Balakrishnan, Sanjay Chawla, and Sam Madden. 2018. RoadRunner: improving the precision of road network inference from GPS trajectories. In *Proceedings of the 26th ACM SIGSPATIAL Conference*. 3–12.

[19] Sophia Karagiorgou, Dieter Pfoser, and Dimitrios Skoutas. 2013. Segmentation-based road network construction. In *Proceedings of the 21st ACM SIGSPATIAL Conference*. 460–463.

[20] Colin Kuntzsch, Monika Sester, and Claus Brenner. 2016. Generative models for road network reconstruction. *International Journal of Geographical Information Science* 30, 5 (2016), 1012–1039.

[21] Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017).

[22] Ailin Mao, Christopher GA Harrison, and Timothy H Dixon. 1999. Noise in GPS coordinate time series. *Journal of Geophysical Research: Solid Earth* 104, B2 (1999), 2797–2816.

[23] Gellért Máttyus, Wenjie Luo, and Raquel Urtasun. 2017. Deeproadmapper: Extracting road topology from aerial images. In *Proceedings of the IEEE International Conference on Computer Vision*. 3438–3446.

[24] M Minghini, MA Brovelli, F Frassinelli, et al. 2018. An open source approach for the intrinsic assessment of the temporal accuracy, up-to-dateness and lineage of OpenStreetMap. In *FOSS4G 2018*, Vol. 42. International Society for Photogrammetry and Remote Sensing, 147–154.

[25] Volodymyr Mnih. 2013. *Machine learning for aerial image labeling*. University of Toronto (Canada).

[26] Volodymyr Mnih and Geoffrey E Hinton. 2010. Learning to detect roads in high-resolution aerial images. In *European Conference on Computer Vision*. Springer, 210–223.

[27] Peter Mooney and Padraig Corcoran. 2014. Analysis of Interaction and Co-editing Patterns amongst OSM Contributors. *Transactions in GIS* 18, 5 (2014), 633–659.

[28] Asif Nawaz, Huang Zhiqiu, Wang Senzhang, Yasir Hussain, Izhar Khan, and Zaheer Khan. 2020. Convolutional LSTM based transportation mode learning from raw GPS trajectories. *IET Intelligent Transport Systems* 14, 6 (2020), 570–577.

[29] Paul Newson and John Krumm. 2009. Hidden Markov map matching through noise and sparseness. In *Proceedings of the 17th ACM SIGSPATIAL Conference*. 336–343.

[30] OpenStreetMap contributors. 2021. Planet dump retrieved from https://planet.osm.org . https://www.openstreetmap.org.

[31] Johannes C Paetzold, Suprosanna Shit, Ivan Ezhov, Giles Tetteh, Ali Ertürk, Helmholtz Zentrum Munich, and Bjoern Menze. 2019. clDice—a novel connectivity-preserving loss function for vessel segmentation. In *Medical Imaging Meets NeurIPS 2019 Workshop*.

[32] Mark L Psiaki and Hee Jung. 2002. Extended Kalman filter methods for tracking weak GPS signals. In *Proceedings of the 15th international technical meeting of the satellite division of the Institute of Navigation (ION GPS 2002)*. 2539–2553.

[33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, 234–241.

[34] Sijie Ruan, Cheng Long, Jie Bao, Chunyang Li, Zisheng Yu, Ruiyuan Li, Yuxuan Liang, Tianfu He, and Yu Zheng. 2020. Learning to generate maps from trajectories. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 890–897.

[35] Stefan Schroedl, Kiri Wagstaff, Seth Rogers, Pat Langley, and Christopher Wilson. 2004. Mining GPS traces for map refinement. *Data mining and knowledge Discovery* 9, 1 (2004), 59–87.

[36] Suprosanna Shit, Johannes C Paetzold, Anjany Sekuboyina, Andrey Zhylka, Ivan Ezhov, Alexander Unger, Josien PW Pluim, Giles Tetteh, and Bjoern H Menze. 2020. clDice–a Topology-Preserving Loss Function for Tubular Structure Segmentation. *arXiv preprint arXiv:2003.07311* (2020).

[37] Rade Stanojevic, Sofiane Abbar, Saravanan Thirumuruganathan, Gianmarco De Francisci Morales, Sanjay Chawla, Fethi Filali, and Ahid Aleimat. 2018. Road network fusion for incremental map updates. In *LBS 2018: 14th International Conference on Location Based Services*. Springer, 91–109.

[38] Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M Jorge Cardoso. 2017. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*. Springer, 240–248.

[39] Tao Sun, Zonglin Di, Pengyu Che, Chun Liu, and Yin Wang. 2019. Leveraging Crowdsourced GPS Data for Road Extraction from Aerial Imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 7509–7518.

[40] Adam Van Etten, Dave Lindenbaum, and Todd M Bacastow. 2018. Spacenet: A remote sensing dataset and challenge series. *arXiv preprint arXiv:1807.01232* (2018).

[41] Suyi Wang, Yusu Wang, and Yanjie Li. 2015. Efficient map reconstruction and augmentation via topological methods. In *Proceedings of the 23rd ACM SIGSPATIAL Conference*. 1–10.

[42] Jan Dirk Wegner, Javier Alexander Montoya-Zegarra, and Konrad Schindler. 2015. Road networks as collections of minimum cost paths. *ISPRS Journal of Photogrammetry and Remote Sensing* 108 (2015), 128–137.

[43] OpenStreetMap Wiki. 2019. Vandalism.

[44] Jing Yuan, Yu Zheng, Chengyang Zhang, Wenlei Xie, Xing Xie, Guangzhong Sun, and Yan Huang. 2010. T-drive: driving directions based on taxi trajectories. In *Proceedings of the 18th ACM SIGSPATIAL Conference*. 99–108.

[45] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. 2018. Road extraction by deep residual u-net. *IEEE Geoscience and Remote Sensing Letters* 15, 5 (2018), 749–753.

[46] Yu Zheng, Xing Xie, Wei-Ying Ma, et al. 2010. GeoLife: A collaborative social networking service among user, location and trajectory. *IEEE Data Eng. Bull.* 33, 2 (2010), 32–39.

[47] Lichen Zhou, Chuang Zhang, and Ming Wu. 2018. D-LinkNet: LinkNet with Pretrained Encoder and Dilated Convolution for High Resolution Satellite Imagery Road Extraction. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2018), 192–1924.