DeepDualMapper: A Gated Fusion Network for Automatic Map Extraction Using Aerial Images and Trajectories

Hao Wu,[1,2,3,5*] **Hanyuan Zhang,**[1,2,3,5*] **Xinyu Zhang,**[1,2,3] **Weiwei Sun,**[1,2,3]
Baihua Zheng,[4] **Yuning Jiang**[5]

[1]School of Computer Science, Fudan University
[2]Shanghai Key Laboratory of Data Science, Fudan University
[3]Shanghai Institute of Intelligent Electronics & Systems
[4]Singapore Management University, [5]Bytedance AI Lab
{wuhao5688, hanyuanzhang16, zhangxinyu, wwsun}@fudan.edu.cn,
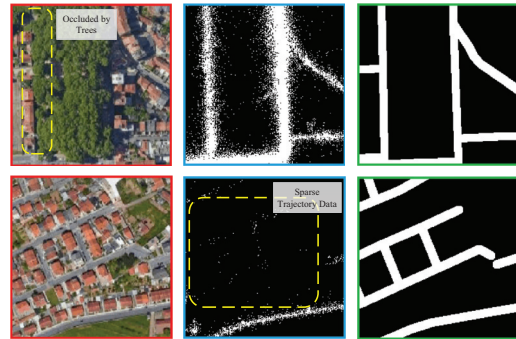bhzheng@smg.edu.sg, jiangyuning@bytedance.com

Abstract

Automatic map extraction is of great importance to urban computing and location-based services. Aerial image and GPS trajectory data refer to two different data sources that could be leveraged to generate the map, although they carry different types of information. Most previous works on data fusion between aerial images and data from auxiliary sensors do not fully utilize the information of both modalities and hence suffer from the issue of information loss. We propose a deep convolutional neural network called DeepDualMapper which fuses the aerial image and trajectory data in a more seamless manner to extract the digital map. We design a gated fusion module to explicitly control the information flows from both modalities in a complementary-aware manner. Moreover, we propose a novel densely supervised refinement decoder to generate the prediction in a coarse-to-fine way. Our comprehensive experiments demonstrate that DeepDualMapper can fuse the information of images and trajectories much more effectively than existing approaches, and is able to generate maps with higher accuracy.

(a) Aerial Image   (b) Trajectory   (c) Map

Figure 1: Examples showing some limitations in aerial images and trajectories (visualized by plotting the GPS points). Eg., roads are covered by trees and trajectories are sparse on minor roads. However, by leveraging both data sources, such problems are expected to be eliminated.

Introduction

Generating city maps is a fundamental building block of many location-based applications like navigation, autonomous vehicle, and so on. Multiple data sources can be leveraged to automatically generate the map. In this paper, we focus on two data sources, namely *aerial images* and *GPS trajectories*. The former is captured by the satellites. Such images are publicly available in many places, eg., Google Earth can stay up-to-date, and have been explored extensively in the past decades for map extraction (Bastani et al. 2018; Zhou, Zhang, and Wu 2018; Sun et al. 2018; Demir et al. 2017, Mnih and Hinton 2010; Cheng et al. 2017). The latter captures human movements in the urban area as most human movements are constrained by the underlying road network. Thanks to the fast development of mobile platforms and localization techniques, collecting these GPS trajectories is much easier nowadays. How to extract the map from trajectory data

is also a hot research problem (Biagioni and Eriksson 2012; Liu et al. 2012b; Chen et al. 2016; Shan et al. 2015).

However, neither data source is perfect for the task of automatic map extraction because of the *information loss* issue. There is inevitably missing information in the data which makes it extremely challenging, if not impossible, to infer the map accurately. Eg., as shown in Fig. 1, in aerial image, some roads may be occluded by trees and buildings; in trajectory data, some less popular roads may have very few, or zero, historical trajectories passing by. Such information loss introduces more complexity to the map extraction process.

As aerial images and trajectory data capture different types of information, the information missed in one source might be available in the other source. In other words, combining two data sources that are complementary offers an effective way to make full use of the information that can be well captured by at least one data source. In the current literature, a variety of researches have tried to fuse aerial images with other auxiliary data sources, including earth observation data including Radar (Fil et al. 2018), Lidar (Audebert, Saux,

and Lefèvre 2018; Piramanayagam et al. 2018; Parajuli et al. 2018), OpenStreetMap data (Audebert, Le Saux, and Lefèvre 2017) and street view images (Cao et al. 2018), to solve the urban scene semantic segmentation problem, which, to some extent, is related to map extraction task. However, these approaches either simply concatenate the features of two modalities (i.e., early-fusion), or compute the average of the predictions made by each modality (i.e., late-fusion). We argue that such rough designs fail to fully explore the utilities of both modalities and are not able to effectively address the issue of information loss faced by both modalities. Among these approaches, the most relevant work solves the same problem by proposing a new deconvolution strategy called *1D decoder* (Sun et al. 2019). However, that work also simply concatenates the image features and the trajectory heat map which leaves room for further improvement.

Consequently, we propose a novel fusion model, namely DeepDualMapper, which aims to fuse the aerial image and trajectory data more seamlessly. We design a *Gated Fusion Module* (GFM) to explicitly learn the modality selection based on the confidence of each data source. It controls the information flows of aerial images and trajectories by *explicitly* defining a learnable weight to make the fusion process complementary-aware, i.e., giving a higher weight to the data source that is more trustworthy and more valuable to infer the answer. Such a design is consistent with the human decision-making process when we need to choose one between two data sources with different confidences. We also introduce a refinement process, conducted through the *Densely Supervised Refinement* (DSR) strategy, to refine the prediction generated by GFM from coarse to fine via residual refinement learning.

In summary, we have made three major contributions in this paper. First, we propose a novel data fusion model called DeepDualMapper which utilizes both aerial images and trajectory data more effectively for the task of map extraction. Second, we design a novel gated fusion module and a refinement decoder that can adaptively control the information flow of both modalities and select the one being more reliable in a coarse-to-fine refinement manner. Such a design follows the heuristics of the human decision-making process when judging two data sources. Third, our model not only outperforms the baselines in three real datasets but also demonstrates a superior resilience to information loss as it can generate the map with higher accuracy than all the existing competitors even when both modalities have 25% information loss.

Related work

Map Extraction via Aerial Images. Many approaches have been proposed to extract a digital map from aerial images. (Mnih and Hinton 2010) uses a feed forward neural network to detect the roads with unsupervised pre-training. After CNN has demonstrated its powerful predictability on visual tasks, many approaches adopt CNNs on this task, such as (Alshehhi et al. 2017) (with FCN), (Cheng et al. 2017) (with DeconvNet), and (Demir et al. 2017) (with SegNet). (Máttyus, Luo, and Urtasun 2017) and (Bastani et al. 2018) are two representative approaches proposed recently for extracting road network topology instead of directly extracting

a binary representation of the map. (Zhou, Zhang, and Wu 2018) and (Sun et al. 2018) adopt U-Net-like structures to generate the maps. These approaches purely rely on the aerial image which has its limitations (eg.roads covered by trees and shadows are not visible) and require high-quality optical sensors.

Map Extraction via Trajectory Data. Approaches leveraging trajectory data mainly rely on clustering techniques to cluster the trajectories or GPS points located on the same road and then to extract a road from each cluster. (Schroedl et al. 2004) performs k-means style clustering on GPS samples to generate road. (Cao and Krumm 2009) clusters the trajectories into a growing road network. (Biagioni and Eriksson 2012) is a representative method transforming trajectories into a discretized image and adopts kernel density estimation to extract the road network. Among these approaches, (Liu et al. 2012b) is reported as the one achieving the best performance (Liu et al. 2012a). Recently, *map update* has also attracted some attentions, which is to incrementally discover new roads and add them to the existing road network (Wang et al. 2013; Shan et al. 2015; Wang, Mao, and Jin 2017; Wu et al. 2015; Chen et al. 2016). In general, these trajectory-based approaches solve the problem in heuristics but not learning-based manner, which makes them inevitably inferior to those learning-based approaches (will be shown in experiments).

Data Fusion for Aerial Images. In the literature, there are some research works using aerial images with the assistance of other remote sensing data sources to reinforce the result. Semantic labeling of urban areas is a representative application for fusion of aerial images and other sensors such as the Lidar point cloud sensor data (Audebert, Saux, and Lefèvre 2018; Piramanayagam et al. 2018), laser data (Audebert, Le Saux, and Lefèvre 2016), OpenStreetMap data (Audebert, Le Saux, and Lefèvre 2017) and street view images (Cao et al. 2018). Late fusion strategy is adopted by (Audebert, Le Saux, and Lefèvre 2017; Audebert, Le Saux, and Lefèvre 2016; Audebert, Saux, and Lefèvre 2018; Piramanayagam et al. 2018). (Audebert, Le Saux, and Lefèvre 2016) improves it by a residual correction strategy called SegNet-RC. Encoder fusion strategy is applied in (Audebert, Le Saux, and Lefèvre 2017; Audebert, Saux, and Lefèvre 2018; Cao et al. 2018; Piramanayagam et al. 2018) via FuseNet (Hazirbas et al. 2016). V-FuseNet, an advanced version, is proposed in (Audebert, Saux, and Lefèvre 2018). Besides, (Fil et al. 2018) uses multiple sensors including satellite image to segment the flooded buildings. It fuses multiples sensors through late fusion strategy. (Parajuli et al. 2018) and (Sun et al. 2019) are the only existing works for fusing aerial images with other auxiliary data source to detect the roads. The former fuses the RGB image with Lidar data while the latter fuses the image with trajectory data to generate the map. However, most of these works either simply concatenate the features or average the predictions of both modalities to perform data fusion and hence have not fully utilized both data sources.
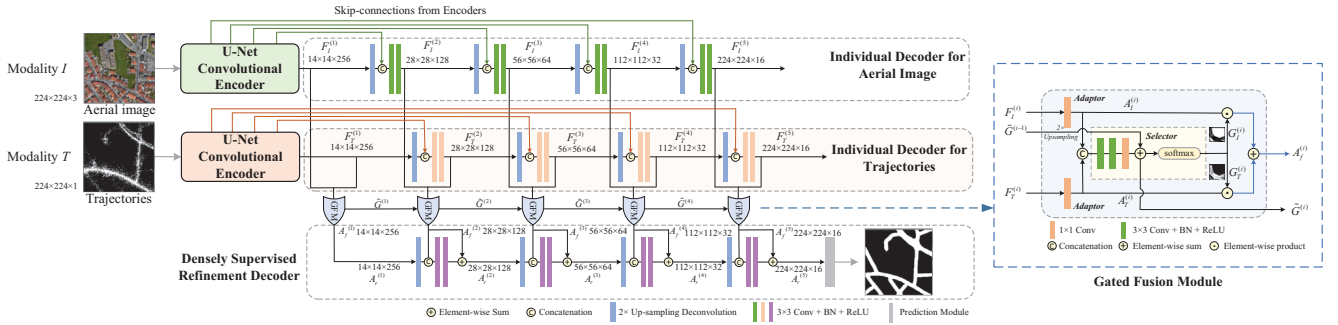
Figure 2: The main fusion architecture of our model. The right part illustrates the details of the gated fusion module. We omit the details of the encoder as it is the same as U-Net. Note that all dense supervisions do not appear in the figure due to the clarity of the image. Please refer to Fig. 5 for the details. All annotated dimensions are assumed by feeding the input in the size of 224×224, for a clearer understanding of each layer. Best viewed in color.

Deep Dual Mapper

Model Overview

We follow the flow of image-based map extraction approaches to transfer the map extraction task into a pixel-wise binary prediction task (Zhou, Zhang, and Wu 2018; Sun et al. 2018; Cheng et al. 2017; Sun et al. 2019). We select U-Net as the main framework for its excellent performance in semantic segmentation, especially biomedical image segmentation, with a simple and elegant structure (Ronneberger, Fischer, and Brox 2015). For the input data, we crop the whole area by 224×224 grids and the trajectory feature is represented by counting the historical GPS points in each pixel. Please refer to the supplementary material for more details on the model and data preprocessing procedure.

Fig. 2 illustrates the main architecture of DeepDualMapper. It consists of two independent U-Nets with each taking in a branch of data source as the input. The original decoders of those two U-Nets serve as the auxiliary decoders to preserve the key information of aerial images and that of trajectories respectively. We extract the activation maps of two branches w.r.t.all scales, i.e., $F_I^{(1)} \sim F_I^{(5)}$ and $F_T^{(1)} \sim F_T^{(5)}$ and fuse them through the *gated fusion module* which produces the fused information $A_f^{(1)} \sim A_f^{(5)}$. Then, we propose the main decoder, the *densely supervised refinement decoder*, which shares the same structure as U-Net's decoder. It takes the fused features $A_f^{(i)}$ as input and generates the pixel-wise prediction. Finally, it generates a feature map that shares the same spatial dimension as the original input dimension and adopts a linear binary predictor to generate pixel-wise predictions. In the following sections, we will detail the gated fusion module and the densely supervised refinement decoder.

Gated Fusion Module

One of the key novelties of our model is the Gated Fusion Module (GFM), which is proposed to simulate the human decision-making process. Given an analytic task and two data sources, we normally select the data source that is more informative and provides more valuable input for the given task. That is to say, for the task of map extraction, we prefer the data source that makes the inference of the roads much

easier and provides more useful information to the other one. To be more specific, it takes the activation maps from two modalities as inputs, i.e., $F_I^{(i)}$ for aerial image and $F_T^{(i)}$ for trajectory where i refers to the level ($1 \sim 5$), and outputs the fused features $A_f^{(i)} = GFM^{(i)} \left(F_I^{(i)}, F_T^{(i)} \right)$. GFM is composed of two sub-modules, i.e., adapter and selector, as detailed in the following.

As the two activation maps are extracted from two independent branches and there is a potential space inconsistency between two modalities, directly fusing them is not an ideal solution. Accordingly, the adapters are introduced to transfer the activation spaces $\mathcal{F}_I$ and $\mathcal{F}_T$ to a uniform space $\mathcal{A}$ to enable a linear combination of the two activation maps in such a uniform space. We denote the adapted features as $A_I^{(i)} = a_I \left(F_I^{(i)} \right) \in \mathbb{R}^{h^{(i)} \times w^{(i)} \times c^{(i)}}$ and $A_T^{(i)} = a_T \left(F_T^{(i)} \right) \in \mathbb{R}^{h^{(i)} \times w^{(i)} \times c^{(i)}}$. Here, $a_I(\cdot)$ and $a_T(\cdot)$ are the channel dimension-preserving adapter operations $\mathbb{R}^{c^{(i)}} \longmapsto \mathbb{R}^{c^{(i)}}$ implemented by the 1×1 convolution, i.e., the channel dimension is linearly transferred into a uniform space $\mathcal{A}$. $h(i)$, $w(i)$, and $c(i)$ refer to the height, width and channel dimension of the feature map respectively. The activation maps are in a uniform space after the transformation, and they can be safely fused by a linear combination, i.e.,

$$A_f^{(i)} = G_I^{(i)} \odot A_I^{(i)} + G_T^{(i)} \odot A_T^{(i)}, \quad s.t., \quad G_I^{(i)} + G_T^{(i)} = \mathbf{1},$$

where $G_I^{(i)} \in \mathbb{R}^{h^{(i)} \times w^{(i)}}$ and $G_T^{(i)} \in \mathbb{R}^{h^{(i)} \times w^{(i)}}$ refer to the gate values w.r.t. aerial image and trajectory respectively; $\odot$ denotes the element-wise product. The complementary constraint $G_I^{(i)} + G_T^{(i)} = \mathbf{1}$ enforces the module to learn a complementary-style fusion. It implies that if certain area in one modality has higher confidence in generating the prediction, it will be assigned a larger weight which simultaneously reduces the weight of the other modality with lower confidence in prediction. If both modalities have sufficient useful information to make a good prediction, the exact values of their weights are not important.

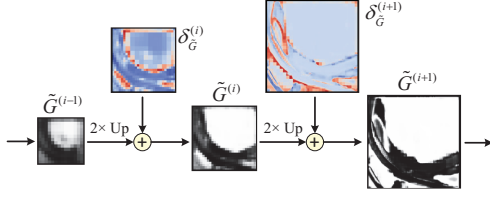The selector sub-module computes the gate values $G_I^{(i)}$

Figure 3: An example showing the coarse-to-fine gate refinement procedure.

Figure 4: An example showing the process of refinement.

and $G_T^{(i)}$ based on the information of two modalities, i.e., $A_I^{(i)}$ and $A_T^{(i)}$. To get the values of $G_I^{(i)}$ and $G_T^{(i)}$, we first compute the un-normalized predictions $\tilde{G}^{(i)} \in \mathbb{R}^{h^{(i)} \times w^{(i)} \times 2}$. $\tilde{G}_I^{(i)} = \tilde{G}^{(i)}[:, :, 0]$ and $\tilde{G}_T^{(i)} = \tilde{G}^{(i)}[:, :, 1]$ are extracted from the slices of $\tilde{G}^{(i)}$. Softmax function is adopted to normalize the gate value w.r.t. each location in $h^{(i)} \times w^{(i)}$ to meet the complementary constraint $G_I^{(i)} + G_T^{(i)} = \mathbf{1}$ as shown in Eq. (2). The un-normalized gates $\tilde{G}^{(i)}$ is computed in a recursive manner formulated in Eq. (1), which means the decision of the gate at level i is based on the decision made at level $(i-1)$ with some residual refinement $\delta_{\tilde{G}}^{(i)}$. The intuition is that we want to perform a consistent regularization on the gate decision space across all scales, i.e., the gates generated from different resolution granularities should not differ much, as the example shown in Fig. 3. Note that $\tilde{G}^{(i-1)}$ has its scale different from $\tilde{G}^{(i)}$, which can be solved by a $2\times$ non-parametric up-sampling operation $U_{2\times}(\cdot)$. For the residual refinement $\delta_{\tilde{G}}^{(i)}$, it is fed by the concatenation of $A_I^{(i)}$ and $A_T^{(i)}$ (i.e., ⓒ denotes the concatenation of two feature maps along the channel dimension), followed by two 3×3 convolution blocks, denoted by $\phi_{3\times3}^2$, with batch normalization and ReLU activation. We adopt the stack of 3×3 convolutions in order to enlarge the receptive field and to collect more global information from those two modalities. Hence, it can make more accurate decisions by a pixel-wise linear transformation which is implemented by an 1×1 convolution $\psi_{1\times1}$ with the output channel at 2. The whole computation flow of the GFM is illustrated in the right part of Fig. 2.

$$\delta_{\tilde{G}}^{(i)} = \psi_{1\times1}\left(\phi_{3\times3}^2\left(A_I^{(i)} ⓒ A_T^{(i)}\right)\right) \in \mathbb{R}^{h^{(i)} \times w^{(i)} \times 2}$$

$$\tilde{G}^{(i)} = U_{2\times}\left(\tilde{G}^{(i-1)}\right) + \delta_{\tilde{G}}^{(i)} \in \mathbb{R}^{h^{(i)} \times w^{(i)} \times 2} \quad (1)$$

$$G_I^{(i)} = \frac{\exp \tilde{G}_I^{(i)}}{\exp \tilde{G}_I^{(i)} + \exp \tilde{G}_T^{(i)}}, G_T^{(i)} = \frac{\exp \tilde{G}_T^{(i)}}{\exp \tilde{G}_I^{(i)} + \exp \tilde{G}_T^{(i)}} \quad (2)$$

Densely Supervised Refinement Decoding

The GFM aims to simulate the human decision-making process for judging the information from two modalities. In this section, we propose a densely supervised refinement decoder (DSRD) to further process the fused information into the prediction map. We introduce it through following two operations, i.e., the fusion refinement and the dense supervision.
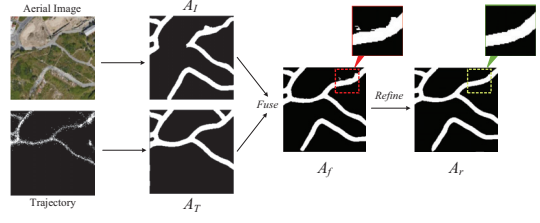
Fusion Refinement. Recall that GFM outputs the fused features $A_f^{(i)}$ that contain the more useful information from two data sources which are expected to be more confident in making the prediction. Inspired by the residual refinement learning (Islam et al. 2017), here, we directly leverage such information as the base of the feature map and we want the decoder to learn a residual refinement of $A_f^{(i)}$ which can be formalized as the following equation.

$$A_r^{(i)} = A_f^{(i)} + R\left(A_f^{(i)}, A_r^{(i-1)}\right)$$

$A_r^{(i)}$ denotes the refined features of level i, based on $A_f^{(i)}$. Here, $R\left(A_f^{(i)}, A_r^{(i-1)}\right)$ is the residual refinement function. Fig. 4 shows an example of the generation of a prediction. First, both image and trajectory modalities make their own independent predictions (with information contained by $A_I^{(i)}$ and $A_T^{(i)}$ respectively), while these predictions might not be sufficient to provide a precise prediction. Next, the GFM fuses these two data sources by enlarging the one being more confident to predict the answer which leads to a more precise prediction. However, we can observe from Fig. 4 that such a linear combination may still face some issues, eg., isolate tiny roads and the un-smoothed predictions. Then, the decoder utilizes the fused feature $A_f^{(i)}$ as the base and learns some area-invariant refinements, such as smoothing the prediction and removing the isolate points/short branches. Such a fusion-refinement process is consistent with our intuition.

The refinement decoder still follows the elegant structure of U-Net, i.e., 4 decoding blocks composed by a stride 2 deconvolutional layer $D_{2\times}(\cdot)$ for learnable $2\times$ up-sampling and two 3×3 convolutions (with batch normalization and followed by an ReLu activation), denoted as $CBR_{3\times3}^2(\cdot)$. Concatenation is adopted to combine the information up-sampled from the previous level ($D_{2\times}\left(A_r^{(i-1)}\right)$) and the information from the current level ($A_f^{(i)}$). The difference is that U-Net concatenates the activation maps from encoders by skip-connection while the newly proposed DSRD concatenates the fused features $A_f^{(i)}$ produced by GFM. The following equation formalizes the residual refinement function. Fig. 5 shows the computation flow of the decoder at i-th level.

$$R\left(A_f^{(i)}, A_r^{(i-1)}\right) = CBR_{3\times3}^2\left(D_{2\times}\left(A_r^{(i-1)}\right) ⓒ A_f^{(i)}\right)$$
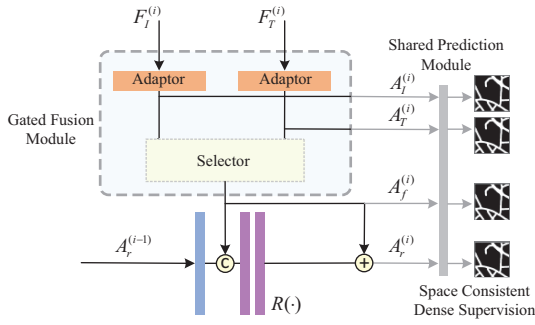
Figure 5: Structure of the densely supervised refinement

Dense Supervision. We want to highlight that, although we design a refinement structure which seems to be reasonable, we could not rely on minimizing the final prediction loss as the only objective. If we do so, the refinement structure might not be able to achieve the performance it is designed to achieve, as there is no guidance to ensure that the decoder has indeed learned how to fuse both modalities and how to refine the prediction. Thus, to enforce our refinement procedure to work as expected, we propose a shared prediction module, which can incorporate dense supervisions to explicitly regularize the learned features and to guide the learning procedure.

Recall that we have introduced two adapters $A_I(\cdot)$ and $A_T(\cdot)$ in GFM to transfer the feature maps $F_I^{(i)}$ and $F_T^{(i)}$ from spaces $\mathcal{F}_I$ and $\mathcal{F}_T$ to a uniform space $\mathcal{A}$. Consequently, the fused feature $A_f^{(i)} = G_I^{(i)} \odot A_I^{(i)} + G_T^{(i)} \odot A_T^{(i)}$, which is the linear combination of $A_I^{(i)}$ and $A_T^{(i)}$, will also lie in $\mathcal{A}$. Besides, the refined feature $A_r^{(i)}$ which is based on $A_f^{(i)}$ (with a learned residual) should also be located in space $\mathcal{A}$.

Accordingly, we decide to adopt a shared prediction module that takes the features in space $\mathcal{A}$ as input and produces a binary prediction. Then, we simultaneously predict the labels from $A_I^{(i)}$, $A_T^{(i)}$, $A_f^{(i)}$ and $A_r^{(i)}$, which involves $5 \times 4 = 20$ supervisions in total. The prediction module takes a tensor $A \in \mathcal{A}$ having shape $h^{(i)} \times w^{(i)} \times c^{(i)}$ as the input and generates a binary prediction with the same spatial size, i.e., $h^{(i)} \times w^{(i)}$. It is implemented by an 1×1 convolutional layer $\zeta(\cdot)$ to perform the linear transformation and a 2-classes softmax layer to predict the probability.

The dense supervision has two functions. First, it explicitly constraints space consistence. Recall that we have only *assumed* that the features produced by the adapters lie in the same space without any constraints in the previous sections. However, after adopting the shared prediction module that only accepts one feature space to generate the prediction, the learned space after the adaption and the space of $A_f^{(i)}$ and $A_r^{(i)}$ are forced to be transferred to the uniform space $\mathcal{A}$. Second, it optimizes the predictions generated by the fused feature $A_f^{(i)} = G_I^{(i)} \odot A_I^{(i)} + G_T^{(i)} \odot A_T^{(i)}$, which is equivalent to optimize the gate values $G_I^{(i)}$ and $G_T^{(i)}$. This supervision actually allows the gradient to be propagated by a short cut

to the GFM to facilitate the learning of the fusion process.

Training. We adopt pixel-wise cross entropy loss on all the predictions in all levels, i.e., predictions computed from $A_I^{(i)}$, $A_T^{(i)}$, $A_f^{(i)}$ and $A_r^{(i)}$, where $i = 1, 2, ..., 5$. We train the model via Adam optimization (Kingma and Ba 2014) with the learning rate at 1e-4. As our model involves many supervisions, the weights of supervisions $A_I^{(i)}$, $A_T^{(i)}$, $A_f^{(i)}$ are all set to 0.5 and that of $A_r^{(i)}$ is set to 1.0 (in all levels). Note that the performance is not very sensitive to the weights. At inference time, $A_r^{(5)}$ is used as the model output. Please refer to the supplementary material for more training details.

Experiments

Experiment Setting

We use three real city-scale datasets containing aerial images and GPS trajectories for evaluation (Porto, Shanghai, and Singapore). Following (Sun et al. 2019; Sun et al. 2018; Zhou, Zhang, and Wu 2018; Parajuli et al. 2018), we adopt intersection of union (IoU) and F1-score of road pixels as the evaluation metrics. We denote correctly predicted road pixels, i.e., true positive, as TP, correctly predicted non-road pixels, i.e., true negative, as TN, road pixels which are wrongly predicted as non-roads, i.e., false negative, as FN, and non-road pixels which are wrongly predicted as roads, i.e., false positive, as FP. The IoU is computed as $\frac{TP}{TP+FP+FN}$. Precision is computed as $\frac{TP}{TP+FP}$ and recall is computed as $\frac{TP}{TP+FN}$. F1-score, the harmonic mean of precision and recall, is computed as $\frac{2 \times precision \times recall}{precision+recall}$. The details of the datasets and the metrics are presented in the supplementary material.

Baselines. We incorporate a large number of competitors which can be clustered into three categories. The first category is trajectory-based approaches that extract the map based on clustering trajectories/GPS points. Representative works *TC1* (Liu et al. 2012b), *KDE* (Biagioni and Eriksson 2012) and *COBWEB* (Shan et al. 2015) are selected as the baselines. The second category is aerial image-based approaches which extract the map using computer vision techniques. *DeconvNet* (Noh, Hong, and Han 2015) (adopted in (Cheng et al. 2017)), *SegNet* (Badrinarayanan, Kendall, and Cipolla 2017) (adopted in (Demir et al. 2017)), *U-Net* (Ronneberger, Fischer, and Brox 2015), *DeepRoadMapper* (Máttyus, Luo, and Urtasun 2017) and *RoadTracer* (Bastani et al. 2018) are implemented as the competitors. We also include the U-Net which takes in trajectory data, pre-processed as an image, as input, denoted as *U-Net$_{traj}$*. The last category is fusion-based approaches which take both aerial image and trajectory as inputs. We implement *Early-fusion* which concatenates the inputs (using U-Net), *L3Fsn* concatenating the features in the third convolutional block of FCN-8 (Piramanayagam et al. 2018), *Late-fusion* and the version with residual correction namely *SegNet-RC*(Audebert, Le Saux, and Lefèvre 2016), and *FuseNet* (Hazirbas et al. 2016) as well as its advanced version *V-FuseNet* applied in (Audebert, Saux, and Lefèvre 2018; Cao et al. 2018; Audebert, Le Saux, and Lefèvre 2017). *TriSeg* (Parajuli et al. 2018) and *1D Decoder* (Sun et al. 2019),

Table 1: The performance of all approaches under three image-trajectory-paired datasets. TC1, KDE and COBWEB are not learning-based approaches thus their results are deterministic.

Dataset	Porto		Shanghai		Singapore	
Metric	IoU	F1-score	IoU	F1-score	IoU	F1-score
Trajectory-based Approaches						
TC1	0.153 ± 0.000	0.184 ± 0.000	0.361 ± 0.000	0.531 ± 0.000	0.109 ± 0.000	0.197 ± 0.000
KDE	0.362 ± 0.000	0.532 ± 0.000	0.333 ± 0.000	0.500 ± 0.000	0.240 ± 0.000	0.386 ± 0.000
COBWEB	0.277 ± 0.000	0.434 ± 0.000	0.334 ± 0.000	0.501 ± 0.000	0.196 ± 0.000	0.328 ± 0.000
U-Net$_{traj}$	0.643 ± 0.010	0.783 ± 0.007	0.498 ± 0.117	0.657 ± 0.109	0.552 ± 0.024	0.711 ± 0.020
Aerial Image-based Approaches						
DeepRoadMapper	0.630 ± 0.007	0.773 ± 0.005	0.481 ± 0.017	0.650 ± 0.016	0.403 ± 0.015	0.575 ± 0.015
RoadTracer	0.658 ± 0.012	0.794 ± 0.009	0.512 ± 0.023	0.677 ± 0.020	0.475 ± 0.013	0.644 ± 0.012
SegNet	0.629 ± 0.006	0.772 ± 0.005	0.484 ± 0.029	0.652 ± 0.026	0.408 ± 0.018	0.578 ± 0.018
DeconvNet	0.589 ± 0.010	0.741 ± 0.008	0.450 ± 0.011	0.617 ± 0.010	0.265 ± 0.022	0.418 ± 0.028
U-Net	0.642 ± 0.013	0.782 ± 0.010	0.479 ± 0.024	0.647 ± 0.022	0.409 ± 0.019	0.580 ± 0.019
Fusion-based Approaches						
Early Fusion	0.664 ± 0.020	0.798 ± 0.015	0.573 ± 0.027	0.728 ± 0.021	0.566 ± 0.031	0.723 ± 0.025
Late Fusion	0.674 ± 0.005	0.806 ± 0.003	0.553 ± 0.027	0.712 ± 0.022	0.561 ± 0.027	0.718 ± 0.022
SegNet-RC (late fusion + correction)	0.688 ± 0.007	0.815 ± 0.005	0.603 ± 0.034	0.752 ± 0.027	0.565 ± 0.042	0.721 ± 0.035
FuseNet (encoder fusion)	0.667 ± 0.016	0.800 ± 0.011	0.558 ± 0.072	0.714 ± 0.062	0.599 ± 0.024	0.749 ± 0.019
V-FuseNet (encoder fusion)	0.688 ± 0.021	0.815 ± 0.015	0.626 ± 0.019	0.770 ± 0.015	0.605 ± 0.029	0.754 ± 0.023
L3Fsn	0.675 ± 0.003	0.806 ± 0.002	0.603 ± 0.012	0.752 ± 0.009	0.555 ± 0.032	0.713 ± 0.027
1D decoder	0.664 ± 0.037	0.797 ± 0.034	0.610 ± 0.018	0.759 ± 0.008	0.599 ± 0.016	0.748 ± 0.017
TriSeg	0.682 ± 0.005	0.811 ± 0.004	0.621 ± 0.005	0.767 ± 0.004	0.589 ± 0.009	0.742 ± 0.007
DeepDualMapper	$\mathbf{0.717 \pm 0.003}$	$\mathbf{0.835 \pm 0.002}$	$\mathbf{0.634 \pm 0.012}$	$\mathbf{0.776 \pm 0.009}$	$\mathbf{0.620 \pm 0.025}$	$\mathbf{0.765 \pm 0.019}$

the only two existing state-of-the-art fusion approaches for map extraction task, are also implemented as the competitors.

Overall Evaluation

We evaluate the performance of different approaches in extracting the maps from three cities under test regions. The results are reported in Table 1. First, most trajectory-based approaches do not perform well. This is consistent with our expectation, as they are based on trajectory clustering without any supervised learning procedure. U-Net$_{traj}$ performs best among them as it is trained end-to-end. For those image-based approaches, DeepRoadMapper and RoadTracer perform better than the others in general as they use specifically designed CNN structure which is more powerful than other VGG-like models. In general, fusion-based approaches outperform the approaches in the above two categories. It demonstrates the power of combining different data sources and justifies that the combination of data sources that complement each other can significantly improve the overall performance. Deep-DualMapper demonstrates its superior performance, as it consistently outperforms all the competitors in all three cases. Note that DeepDualMapper achieves a much stabler performance as its standard deviation tends to be small. Note that to prove the effectiveness of our fusion strategy and to assure the fairness of our evaluation, all the approaches evaluated use the same input features. Among other fusion-based approaches, TriSeg, 1D decoder, and V-FuseNet demonstrate certain advantages over others; while the performance ranking of remaining approaches is not clear.

Notice that among these three datasets, the performance on the Porto dataset is the best. The main reason is that Porto dataset has better data quality than others. Porto's satellite image is more accurate and clearer than the other two. For trajectory data, the noise and error on trajectories of Porto are smaller than that of Singapore, and the coverage density

is higher than that of Shanghai. Consequently, the overall performance of Porto is the best of all three datasets.

Visualization of the Gate Values

To offer a clearer view of how our fusion model behaves when it encounters two data sources with different confidence of predictability, we select a sub-region and gradually reduce the predictability in either modality. We report the gate values $G_I^{(5)}$ and $G_T^{(5)}$ of level 5 (i.e. the last level) in Fig. 6 to visualize the changes to the information flow when one data source's quality drops. We can infer that when the image fades and starts to lose more and more details, it gradually loses its confidence in prediction. Therefore, $G_I^{(5)}$ is reduced (becomes blacker) while $G_T^{(5)}$ is increased (becomes whiter). However, it is still able to make a reasonable prediction even when the original aerial image has lost much important information. Similarly, we can make the same observations from the case where the noise of trajectory data increases.

Performance w.r.t. Trajectory Data Volume

Fig. 8 plots the IoU performance of U-Net$_{traj}$ and Deep-DualMapper under different trajectory data volumes. We only report the results of the Porto dataset for space saving, while similar results are observed from the other two datasets. U-Net$_{traj}$ is selected for comparison because it is the backbone of our model and it can be directly compared with DeepDualMapper that takes in multiple data sources as input. It is observed that U-Net$_{traj}$ is vulnerable to the reduction of trajectory data quality, while DeepDualMapper demonstrates resilience to the reduction of data quality. Although Deep-DualMapper is relatively robust to data quality, it still has a minor performance drop. We think it is reasonable. When one modality (i.e.. trajectory data) loses most of its information, DeepDualMapper can only infer the information from
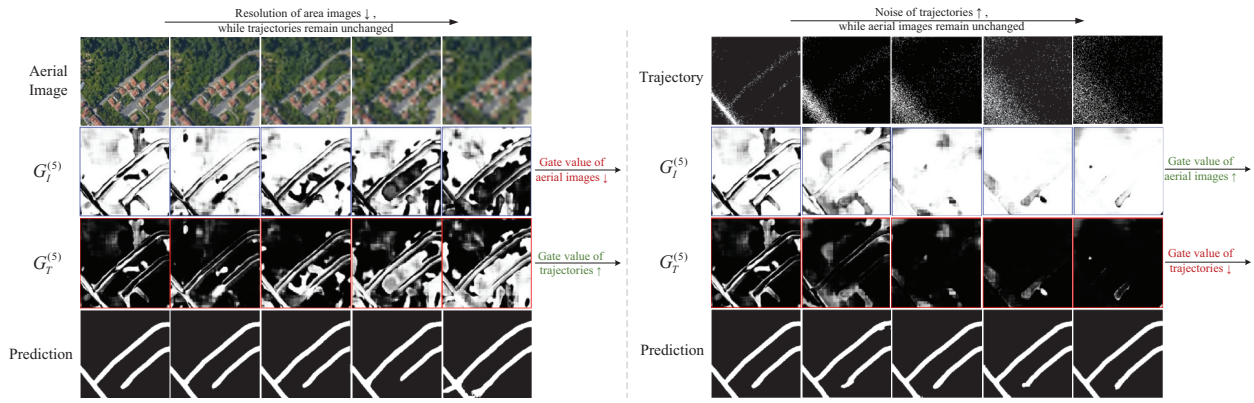
Figure 6: The visualization shows how gate value changes w.r.t. one data source being less confident in prediction. The left sample refers to the case where the resolution of an image is reduced gradually, and the right one refers to the case where the localization noises of trajectories are increased. White pixel indicates a large gate value and black pixel indicates a small value.
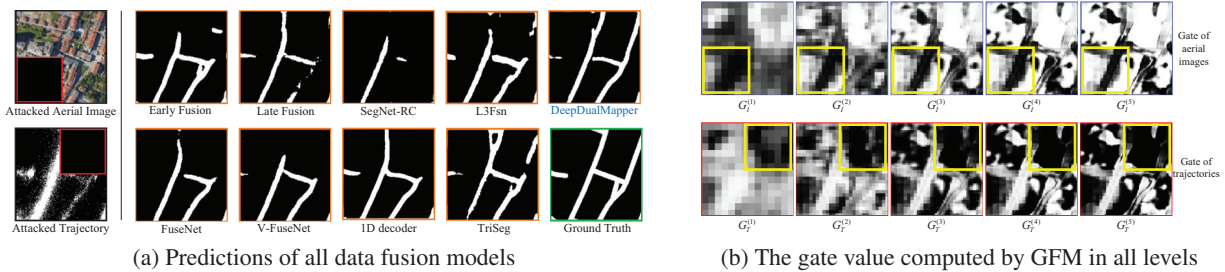

(a) Predictions of all data fusion models   (b) The gate value computed by GFM in all levels

Figure 7: Visualization of results and gate values under information loss attack. White pixel indicates higher gate value. Pay attention to the areas (in yellow boxes) of gate values which represent the areas attacked by the information loss.
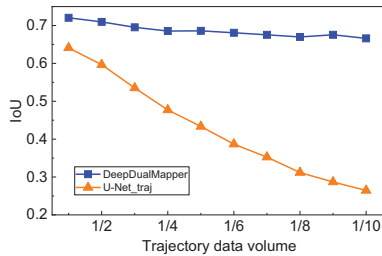

Figure 8: The IoU of DeepDualMapper (dual modality) and U-Net$_{traj}$ (single modality) under Porto dataset w.r.t. different trajectory data qualities.

the other modality (i.e., aerial image) with its performance similar to the U-Net of single modality (i.e., aerial image).

Evaluation on Robustness to Information Loss

Data fusion approaches will be effective only if they own the capability of learning how to compensate for the information loss of one modality by other modalities. As stated before, existing approaches roughly fuse the data sources so it remains unknown whether they have really learned how to correctly fuse the data and to maximize the utility of both data sources. To have a more thorough study, we conduct the information loss attack on *testing* datasets. In detail, we ran-

domly remove the information of 1/4 area of the aerial image and another 1/4 area of the trajectory data, to see whether the fusion model can correctly find out the right modality containing the true information to perform the inference. We conduct the experiment in all three datasets and report the quantitative results in Table 2. In addition, we visualize the maps generated by all data fusion models in Fig. 7(a). Note that quadrant 3 of the aerial image and quadrant 1 of the trajectory data have been removed. Most of the baselines are vulnerable to the information loss attack while our model, to some extent, is able to defend such attack.

To further study how DeepDualMapper fuses the data and why it outperforms the other approaches under an information loss attack, we visualize the gate values computed by GFM from level 1 to level 5 in Fig. 7(b). We can observe that since quadrant 3 of the aerial image has been removed, GFM assigns higher weights to the trajectory data. Similarly, as quadrant 1 of the trajectory data has been removed, GFM detects that the aerial image carries more valuable information for the prediction. Accordingly, it passes the information of aerial image but blocks the trajectory information in quadrant 1. This example demonstrates the effectiveness of our gating mechanism for data fusion. In addition, from the gate values across different levels, we can observe that the gate values are refined from coarse to fine and the gate values of all levels meet the consistency constraint.

Table 2: The reported IoU under the test set with information loss attack. Note we do not include the attack in training set.

Dataset	Porto	Shanghai	Singapore
Early Fusion	0.500 ± 0.023	0.440 ± 0.014	0.420 ± 0.016
Late Fusion	0.475 ± 0.011	0.407 ± 0.059	0.379 ± 0.035
SegNet-RC	0.546 ± 0.025	0.442 ± 0.028	0.393 ± 0.026
FuseNet	0.463 ± 0.042	0.455 ± 0.012	0.405 ± 0.039
V-FuseNet	0.530 ± 0.028	0.432 ± 0.043	0.445 ± 0.016
L3Fsn	0.510 ± 0.027	0.450 ± 0.011	0.391 ± 0.045
1D decoder	0.471 ± 0.043	0.455 ± 0.005	0.411 ± 0.037
TriSeg	0.462 ± 0.030	0.467 ± 0.022	0.398 ± 0.024
DeepDualMapper	$\mathbf{0.588 \pm 0.019}$	$\mathbf{0.497 \pm 0.023}$	$\mathbf{0.477 \pm 0.030}$

Study on Densely Supervised Refinement

Recall that the DSRD is designed to be responsible for generating the predictions. We first evaluate the IoU performance of the predictions generated by the features of $A_I^{(5)}$ (image feature), $A_T^{(5)}$ (trajectory feature), $A_f^{(5)}$ (fused feature) and $A_r^{(5)}$ (refined feature) via the shared prediction module. The results are shown in Fig. 9(a). As compared with the information captured by the features $A_I^{(5)}$ or $A_T^{(5)}$ of single modality, the fused feature $A_f^{(5)}$ contains more precise information. In addition, we are able to observe the improvement achieved by $A_r^{(5)}$ over $A_f^{(5)}$, which demonstrates the effectiveness of the refinement process. Last but not least, we claim that supervision can further improve the fusion process as the prediction made by the fused feature A_f with the dense supervision is more accurate than that without the supervision, as reported in Fig. 9(b). For a clearer view, we visualize the predictions computed by the features of $A_I^{(i)}$, $A_T^{(i)}$, $A_f^{(i)}$ and $A_r^{(i)}$ in all 5 levels in Fig. 10. From the results, we can observe that the refinement procedure smooths and connects the roads and the result is generated through the feature maps from coarse to fine as well as from rough to smooth over five levels.

Conclusion

In this paper, we have presented an automatic map extraction approach that can effectively fuse the information of aerial images and that of GPS trajectories. We further boost the performance and accuracy of map extraction task by proposing the gated fusion module and the densely supervised refinement decoder. We have demonstrated the effectiveness of our model through comprehensive experiment studies based on three city-scale datasets. In addition, we have implemented an information loss attack task. As expected, our model is much more robust to the attack, compared with a wide range of state-of-the-art competitors, with its resilience to the information loss mainly contributed by its delicately designed fusion structure. Since our method is still relatively simple for the preprocessing of trajectory data, we plan to consider other information such as speed and direction in the future and investigate whether performing more data augmentation could enhance the performance of the model.

Acknowledgements

(a) IoU predicted by A_I, A_T, A_f and A_r
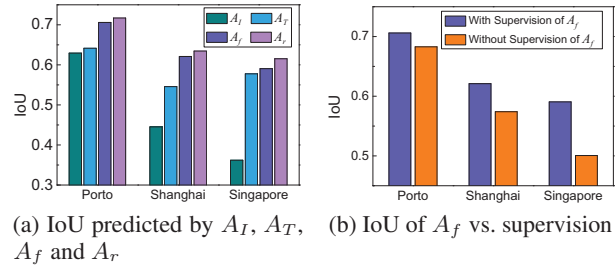
(b) IoU of A_f vs. supervision

Figure 9: Quantitative results showing the utility of densely supervised refinement decoding. The IoU of predictions generated by features at each stage are gradually increased demonstrating the effectiveness of the fusion step and the refinement step.
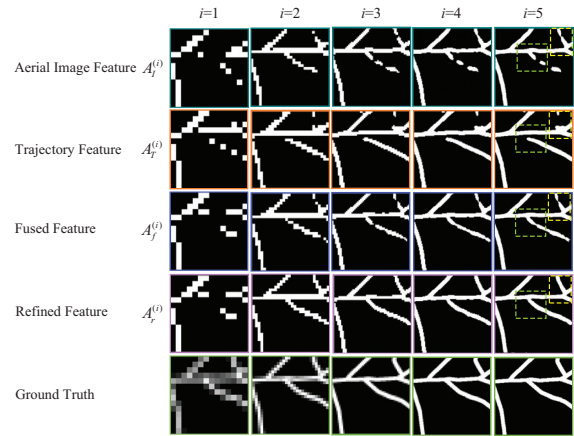

Figure 10: Visualizations of the predictions generated by A_I, A_T, A_f and A_r in all 5 levels. The fused predictions are better than those of A_I and A_T. The refined predictions further smooth (and connect) the roads predicted by A_f. Please pay attention to the differences in the dotted boxes.

References

Alshehhi, R.; Marpu, P. R.; Woon, W. L.; and Dalla Mura, M. 2017. Simultaneous extraction of roads and buildings in remote sensing imagery with convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing* 130:139–149.

Audebert, N.; Le Saux, B.; and Lefèvre, S. 2016. Semantic segmentation of earth observation data using multimodal and multi-scale deep networks. In *ACCV'16*, 180–196.

Audebert, N.; Le Saux, B.; and Lefèvre, S. 2017. Joint learning from earth observation and openstreetmap data to get faster better semantic maps. In *CVPR'17 Workshops*, 1552–1560.

Audebert, N.; Saux, B. L.; and Lefèvre, S. 2018. Beyond RGB: very high resolution urban remote sensing with multimodal deep

networks. *ISPRS Journal of Photogrammetry and Remote Sensing* 140:20–32.

Badrinarayanan, V.; Kendall, A.; and Cipolla, R. 2017. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis & Machine Intelligence* (12):2481–2495.

Bastani, F.; He, S.; Abbar, S.; Alizadeh, M.; Balakrishnan, H.; Chawla, S.; Madden, S.; and DeWitt, D. 2018. Roadtracer: Automatic extraction of road networks from aerial images. In *CVPR'18*.

Biagioni, J., and Eriksson, J. 2012. Map inference in the face of noise and disparity. In *SIGSPATIAL GIS'12*, 79–88.

Cao, L., and Krumm, J. 2009. From gps traces to a routable road map. In *SIGSPATIAL GIS'09*, 3–12.

Cao, R.; Zhu, J.; Tu, W.; Li, Q.; Cao, J.; Liu, B.; Zhang, Q.; and Qiu, G. 2018. Integrating aerial and street view images for urban land use classification. *Remote Sensing* 10(10):1553.

Chen, C.; Lu, C.; Huang, Q.; Yang, Q.; Gunopulos, D.; and Guibas, L. 2016. City-scale map creation and updating using gps collections. In *KDD'16*, 1465–1474.

Cheng, G.; Wang, Y.; Xu, S.; Wang, H.; Xiang, S.; and Pan, C. 2017. Automatic road detection and centerline extraction via cascaded end-to-end convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing* 55(6):3322–3337.

Demir, I.; Hughes, F.; Raj, A.; Tsourides, K.; Ravichandran, D.; Murthy, S.; Dhruv, K.; Garg, S.; Malhotra, J.; Doo, B.; et al. 2017. Robocodes: Towards generative street addresses from satellite imagery. In *CVPR'17 Workshops*, 1486–1495.

Fil, J.; Rudner, T. G.; Russwurm, M.; Bischke, B.; Pelich, R.; Kopackova, V.; and Bilinski, P. 2018. Multi[3]net: Segmenting flooded buildings via fusion of multiresolution, multisensor, and multitemporal satellite imagery. *arXiv preprint arXiv:1812.01756*.

Hazirbas, C.; Ma, L.; Domokos, C.; and Cremers, D. 2016. Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. In *ACCV'16*, 213–228.

Islam, M. A.; Naha, S.; Rochan, M.; Bruce, N.; and Wang, Y. 2017. Label refinement network for coarse-to-fine semantic segmentation. *arXiv preprint arXiv:1703.00551*.

Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Liu, X.; Biagioni, J.; Eriksson, J.; Wang, Y.; Forman, G.; and Zhu, Y. 2012a. Mining large-scale, sparse gps traces for map inference: comparison of approaches. In *KDD'12*, 669–677.

Liu, X.; Zhu, Y.; Wang, Y.; Forman, G.; Ni, L. M.; Fang, Y.; and Li, M. 2012b. Road recognition using coarse-grained vehicular traces. *Hp Labs*.

Máttyus, G.; Luo, W.; and Urtasun, R. 2017. Deeproadmapper: Extracting road topology from aerial images. In *ICCV'17*, 3458–3466.

Mnih, V., and Hinton, G. E. 2010. Learning to detect roads in high-resolution aerial images. In *ECCV'10*, 210–223.

Noh, H.; Hong, S.; and Han, B. 2015. Learning deconvolution network for semantic segmentation. In *ICCV'15*, 1520–1528.

Parajuli, B.; Kumar, P.; Mukherjee, T.; Pasiliao, E.; and Jambawa-likar, S. 2018. Fusion of aerial lidar and images for road segmentation with deep cnn. In *SIGSPATIAL GIS'12*, 548–551.

Piramanayagam, S.; Saber, E.; Schwartzkopf, W.; and Koehler, F. 2018. Supervised classification of multisensor remotely sensed images using a deep learning framework. *Remote Sensing* 10(9):1429.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI'15*, 234–241.

Schroedl, S.; Wagstaff, K.; Rogers, S.; Langley, P.; and Wilson, C. 2004. Mining gps traces for map refinement. *Data mining and knowledge Discovery* 9(1):59–87.

Shan, Z.; Wu, H.; Sun, W.; and Zheng, B. 2015. Cobweb: a robust map update system using gps trajectories. In *UbiComp'15*, 927–937.

Sun, T.; Chen, Z.; Yang, W.; and Wang, Y. 2018. Stacked u-nets with multi-output for road extraction. In *CVPR'18 Workshop*, 202–206.

Sun, T.; Di, Z.; Che, P.; Liu, C.; and Wang, Y. 2019. Leveraging crowdsourced gps data for road extraction from aerial imagery. In *CVPR'19*, 7509–7518.

Wang, Y.; Liu, X.; Wei, H.; Forman, G.; Chen, C.; and Zhu, Y. 2013. Crowdatlas: Self-updating maps for cloud and personal use. In *MobiSys'13*, 27–40.

Wang, T.; Mao, J.; and Jin, C. 2017. Hymu: A hybrid map updating framework. In *DASFAA'17*, 19–33.

Wu, H.; Tu, C.; Sun, W.; Zheng, B.; Su, H.; and Wang, W. 2015. Glue: a parameter-tuning-free map updating system. In *CIKM'15*, 683–692.

Zhou, L.; Zhang, C.; and Wu, M. 2018. D-linknet: Linknet with pre-trained encoder and dilated convolution for high resolution satellite imagery road extraction. In *CVPR'18 Workshop*, 182–186.