

# LEARNING TO INTEGRATE VISION DATA INTO ROAD NETWORK DATA

Oliver Stromann<sup>\*†</sup>   Alireza Razavi<sup>†</sup>   Michael Felsberg<sup>\*</sup>

<sup>\*</sup> Computer Vision Laboratory, Linköping University, Sweden

<sup>†</sup> Scania CV AB, Autonomous Transport Solution Research, Sweden

## ABSTRACT

Road networks are the core infrastructure for connected and autonomous vehicles, but creating meaningful representations for machine learning applications is a challenging task. In this work, we propose to integrate remote sensing vision data into road network data for improved embeddings with graph neural networks. We present a segmentation of road edges based on spatio-temporal road and traffic characteristics, which allows enriching the attribute set of road networks with visual features of satellite imagery and digital surface models. We show that both, the segmentation and the integration of vision data can increase performance on a road type classification task, and we achieve state-of-the-art performance on the OSM+DiDi Chuxing dataset on Chengdu, China.

**Index Terms**— Graph Neural Networks, Remote Sensing, Road Networks

## 1. INTRODUCTION

Knowledge about road networks (RNs) and the traffic flowing through it is key for making good decisions for connected and autonomous vehicles. It is therefore of large importance to find meaningful and efficient representations of RNs to allow and ease the knowledge generation.

However, the structural information encoded in spatial network data like RNs has two shortcomings: first, the incompleteness of the encoded information and second the absence of information that is difficult to encode yet still relevant. On the other hand, there exist vast amounts of unstructured image data that contains complementary data. Therefore, we propose to enrich the structured but incomplete spatial network data with unstructured but spatially complete data in the form of continuous imagery. We demonstrate the integration of vision data into a Graph Neural Network (GNNs) by low-level visual features and propose a segmentation of the road network graph according to spatial and empirical traffic constraints.

---

We gratefully acknowledge the support of Sweden's innovation agency, Vinnova, through project iQDeep (project number 2018-02700). Computations were enabled by resources provided by the SNIC at C3SE partially funded by the Swedish Research Council through grant agreement no. 2018-05973.

Specifically, we address the challenge of enriching crowd-sourced RN data from OpenStreetMap (OSM) [1] with remotely sensed data from Maxar Technologies [2]. We do so by utilizing GPS tracks from DiDi Chuxing's ride hailing service [3] to spatially segment the RN based on empirical travel times. We evaluate our proposed method on a node classification task for road type labels using GraphSAGE [4].

The results confirm that our approach leads to improved performance on a classification of road type labels in both supervised and unsupervised learning settings. To summarize, our contributions are: 1) integrating image data through low-level visual features into a graph representation of spatial structures by means of spatial segmentation 2) a systematic evaluation of our proposed method in a supervised and unsupervised learning of node classifications.

## 2. RELATED WORK

### 2.1. Geographical Data on Road Networks

Graph data on RNs is spatial network data, which is nowadays easily accessed through common mapping sources. The RN is represented as an attributed directed spatial graph  $G=(V, E)$ , with intersections as nodes  $v \in V$  and roads connecting these intersections as edges  $e \in E$ . From the crowd-sourced OSM [1], intersection attributes, and road attributes can be obtained. One shortcoming of the available crowd-sourced RN data is that some attributes are not consistently recorded. Either data is missing completely for certain geographical regions, or it is inconsistently set [5].

Remote sensing data is data collected from air- or space borne sensors like radars, LiDARs or cameras. It requires extensive data preprocessing like atmospheric, radiometric and topographic error correction. Nowadays, analysis-ready remote sensing data is available which has undergone correction.

### 2.2. Machine Learning Concepts

#### 2.2.1. Graph Neural Networks

GNNs experienced a surge in recent years. Many architectures have been proposed to produce deep embeddings of nodes, edges, or whole graphs [6, 7]. Techniques from other deep learning domains such as computer vision and natural language processing have been successfully integrated into GNN architectures [7]. In our work we focus on learning

node embeddings with GraphSAGE [4] - a GNN architecture which relies on an efficient sampling of neighbour nodes and an aggregation of their features.

### 2.2.2. Visual Feature Extraction

Visual feature extraction is the process of obtaining information from images to derive informative, non-redundant features to facilitate subsequent machine learning tasks. Convolutional Neural Networks (CNNs) replaced hand-craft extractors in most applications. CNNs either require data of the application to train or need to be pre-trained on enormous datasets to then be transferred to the application domain [8]. We explicitly evaluate a simple pixel statistic - intensity histograms [9] - in our approach.

## 2.3. Machine Learning on Geographical Data

### 2.3.1. Machine Learning on Road Networks

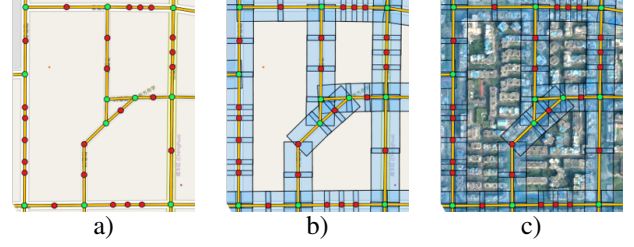
Examples of machine learning tasks that are applied to RNs range method-wise from classification and regression to sequence-prediction and clustering [10, 11, 12], and application-wise from vehicle-centric predictions such as next-turn, destination and time of arrival predictions or routing to RN-centric predictions such as speed limit, travel time or traffic flow predictions [12, 13].

Following the growing popularity of OSM, several machine learning methods have been proposed in the recent years to either improve or to use OSM data [14]. For RNs, conventional machine learning techniques that do not exploit the graph structure have been used to predict road type labels or to impute missing attributes and topologies [5, 15].

GNNs offer the advantage that graph topologies are directly exploited, and no additional features need to be constructed. Consequently, several authors have demonstrated the effectiveness of GNNs on RNs in classifications [11, 12, 16], regressions [11] and sequence predictions [12]. Jepsen *et al.* [11] propose a relational fusion network (RFN), which use different representations of a road network concurrently to aggregate features. Wu *et al.* [12] developed a hierarchical road network representation (HRNR) in which a three-level neural architecture is constructed that encodes functional zones, structural regions and roads respectively. He *et al.* [16] proposed an integration of visual features from satellite imagery through CNNs as node features to a GNN.

### 2.3.2. Visual Feature Extraction in Remote Sensing

As in many other domains, transfer learning has also been applied in remote sensing to address common tasks like scene classification [8] or object detection [17]. On the other hand, hand-crafted features combined with classical machine learning algorithms are still commonly used in computer vision on remote sensing. Object-based land cover classification frequently uses hand-crafted texture or intensity histogram features and still achieve state-of-the-art performance [18, 19, 20].



**Fig. 1.** Processing steps of our proposed method. a) Graph  $G_{ORN}$  with intersections  $V_{ORN}$  in green and roads  $E_{ORN}$  in yellow. Overlaid is  $G_{SRN}$  with interstitial vertices in red. b) Rectangular road segment footprints in blue. c) *TrueOrtho* RGB-imagery. Pixels within each blue rectangle are added to the road segment attributes.

## 3. MATERIALS & METHODS

### 3.1. Road Network Graph

Our datasets consist of RN data, GPS tracks and remote sensing imagery. We follow the commonly used approach of dual graph representation  $G^D=(E, B)$  of RNs [11, 10, 12, 13]. That means, the graph consists of road segments as nodes  $e \in E$  and connection between roads segments as edges  $b \in B$ .

In this study, RN data is obtained from OSM [1] using OSMnx [21]. We add traffic information to the road attributes by matching GPS tracks of ride hailing vehicles to the RN [22].

### 3.2. GraphSAGE

GraphSAGE [4] is a graph convolutional neural network, which for an input graph  $G=(V, E)$  produces node representations  $\mathbf{H}_v^k$  at layer depth  $k$ . This is done by aggregating features from neighbouring nodes  $N(v)$  using an aggregation function  $\text{AGG}^k$  and after a linear matrix multiplication with a weight matrix  $\mathbf{W}^k$  apply a non-linear activation function  $\sigma$ , such that

$$\mathbf{H}_v^k = \sigma \left( \mathbf{W}^k \cdot \text{AGG}^k \left( \{ \mathbf{H}_n^{(k-1)} | n \in N(v) \} \right) \right). \quad (1)$$

At layer depth  $k=0$  the aggregated features consist of the neighbouring nodes' attributes. The node representations  $\mathbf{H}_v^K$  after the last layer (i.e.,  $k=K$ ) can be used as node embeddings  $\mathbf{z}_v$  which serve as input for a downstream machine learning task, like a classifier in our case. The interested reader is referred to Hamilton *et al.* [4] for implementation details of GraphSAGE.

## 4. CONTRIBUTIONS

We propose to enrich the attribute set of spatial network data of RNs with visual features from remote sensing data and present a segmentation of the RN into a fine-grained graph representation. Figure 1 outlines the processing steps of our proposed method.

#### 4.1. Segmentation

While the topological representation of an RN is advantageous for routing applications, we argue that a more fine-grained representation of an RN is better suited for many machine learning tasks. In a topological RN, road geometries may vary in length, and road attributes might be non-static for the entirety of a road. Therefore, we introduce a segmentation which creates a more fine-grained representation of the RN with shorter edges. To take the spatio-temporal nature of traffic on the RN into account, the segmentation aims to create segments of target travel times as well as target segment lengths.

---

#### Algorithm 1 Segmentation Algorithm.

---

```

1: procedure SEGMENT( $G = (V, E)$ ,  $attr$ ,  $target_{attr}$ )
2:    $\triangleright$  Input graph  $G$ , edge attribute  $attr$  and target value  $target_{attr}$ 
3:   for  $(u, v, a) \in E$  do
4:      $n \leftarrow \lceil a_{attr} / target_{attr} \rceil$ 
5:      $\triangleright$  Calculate number of segmentations  $n$ 
6:      $u_1 \leftarrow u$ 
7:     for  $i \leftarrow 1, n$  do
8:        $v_i \leftarrow \text{INTERPOLATE}(a_{geom}, \frac{i}{n})$ 
9:      $\triangleright$  Add node  $v_i$  at  $\frac{i}{n}$ -distance of  $a_{geom}$  between  $u$  and  $v$ 
10:     $V \leftarrow V \cup v_i$ 
11:     $E \leftarrow E \cup (u_i, v_i)$ 
12:     $u_{i+1} \leftarrow v_i$ 
13:  end for
14:   $E \leftarrow E \setminus (u, v)$ 
15:   $V \leftarrow V \setminus (u, v)$ 
16: end for
17: return  $G$ 
18: end procedure

```

---

Given a target segment travel time  $target_{traveltime}$  and a target segment length  $target_{length}$ , we determine the number of equally distanced split points along a road geometry. At these points, interstitial nodes are inserted, replacing original edges with shorter edges. From the original RN  $G_{ORN}$ , the segmented RN  $G_{SRN}$  is thus created by running Algorithm 1 first with  $attr=traveltime$  and secondly with  $attr=length$ . The segmentation is done prior to the conversion of graph  $G$  to its dual representation  $G^D$ .

#### 4.2. Integration of Vision Data

Though the traffic information and segmentation enriches the attribute sets, the structural information in the spatial network of  $G$  still suffers from incompleteness in the encoded information, as not all attributes are consistently set in the crowd-sourced data from OSM. Furthermore, potentially relevant information in the vicinity of the spatial network are not covered in the attribute sets.

Continuous image data on the other hand has the potential to capture such information. Hence, we enrich the

road attribute set with visual features from remote sensing data. Analysis-ready high-resolution orthorectified satellite imagery and a Digital Surface Model (DSM) are used in our study. We use a rectangle around a road footprint (Figure 1 b)) to extract image patches of the RGB imagery and the DSM (Figure 1 c)). From each image patch, we use intensity histograms per channel as visual features. The frequencies in each histogram bin are used as numerical features in the road attributes. These features have the benefit compared to CNNs that they require no additional training.

#### 4.3. Summary

We present a method to enrich structural, but incomplete, spatial network data with less-structured, but spatially complete data in the form of continuous imagery. We rely on simple low-level visual features, that require no learning, and demonstrate the effectiveness of this approach in the context of crowd-sourced RN data and high-resolution remote sensing imagery. Our work is closely related to RoadTagger by He *et al.* [16]. However, RoadTagger relies purely on CNNs to extract visual features and trains them end-to-end with the GNN, whereas we fuse low-level visual features with the road network attributes from OSM. Moreover, we extend the experiments also to unsupervised learning and demonstrate that it can achieve comparable performance on the binary road type classification problem. In general, our proposed method offers a light-weight alternative to CNN-based approaches, while achieving comparable performances.

### 5. EXPERIMENTS

#### 5.1. Datasets

An RN from the city of Chengdu, China is extracted from OSM [1]. We match GPS trajectories from ride-hailing vehicles to the RN. We use high-resolution (0.5 m/pixel) analysis-ready orthorectified satellite imagery (*TrueOrtho*) and a Digital Surface Model (*DSM*) as vision data.

We construct three datasets: The unsegmented, original RN (ORN), the segmented RN (SRN) and the segmented RN visual features (SRN+VIS). In SRN and SRN+VIS, interstitial nodes are inserted according to the proposed segmentation in Algorithm 1. We set the target travel time  $target_{traveltime}$  to 15 s and target segment length  $target_{length}$  to 120 m. In ORN, we randomly sample 20% of the nodes in  $E$  for validation and 20% for testing. The remaining nodes are used for training. Validation and test set allocations are propagated down from ORN to SRN and SRN+VIS.

The following features make up the feature set: Geographical features of *length*, *bearing*, *centroid*, *geometry* which is the road geometry resampled to a fixed number of equally-distanced points and translated by the *centroid* to yield relative distances in northing and easting (meters). Binary features depicting *one-way*, *bridge* and *tunnel*. GPS features from the matched trajectories, consisting of *travel times* as median travel times and *throughput* as average daily

vehicle throughput. Additionally, we extract visual features from image patches of 120 m by 120 m from *TrueOrtho* and *DSM* following the *bearing*. In SRN+VIS the pixel values of each channel (3 RGB channel for the *TrueOrtho*, 1 graylevel channel for *DSM*) are binned into a histogram of 32 bins.

## 5.2. Training

To demonstrate how the RN segmentation and the integration of visual features improve the node embedding, we train for node classifications of road type labels.

The *highway* label from OSM is used as the target. The label describes the type of road as an indicator of road priority. The classes are *motorway*, *trunk*, *primary*, *secondary*, *tertiary*, *unclassified*, *residential* and *living street*. *Living street* and *motorway* are underrepresented in both ORN and SRN with less than 2% of all samples. This class imbalance makes the 8-class classification a challenging problem. Additionally, to set this work into context of RoadTagger [16], we perform a binary classification, by aggregating the 8-class predictions of the first four classes (*motorway* to *secondary*) and the remaining four classes (*tertiary* to *living street*) to two classes.

We train a GraphSAGE model with layer depth  $K = 2$  and mean pooling aggregators using Adam optimizer. Model selection is based on validation performance and reported are test performances. We perform a Bayesian hyperparameter search [23]. The search space of hyperparameters is composed of *hidden units*  $\in \{512, 1024\}$ , *embedding dimensionality*  $\in \{2^3, 2^4, \dots, 2^7\}$ , *learning rate*  $\in [1e-8, 1e-1]$ , *weight decay*  $\in [0, 0.1]$  and *dropout rate*  $\in [0, 0.4]$ . Unsupervised models are trained for 20 epochs with a batch size of 1024. Supervised settings are trained for 100 epochs with a batch size of 512.

## 6. NUMERICAL RESULTS

Tables 1 and 2 show the test results on road type classification in micro-averaged F1-Scores for supervised and unsupervised learning respectively. A majority voting from SRN and SRN+VIS to ORN is stated. The first two rows depict performance on the 8-class classification problem of eight road type labels. The performance on the binary classification is depicted in the last two rows.

The supervised results in table 1, the majority vote of the different SRN subsets shows clearly that SRN+VIS produces the best performing model with a 13.5% improvement compared to ORN. The segmentation alone (SRN) improves the performance only to a small extent of 1.4%. When aggregating the predictions to two classes, SRN+VIS again achieves the best performance with a 0.7% improvement over ORN.

In the unsupervised setting in table 2, the majority voting from SRN is on par with ORN, while SRN+VIS achieves a small improvement of 0.4%. For the binary classification, SRN shows a small performance decrease of 0.7%, while SRN+VIS improves the classification by 0.2%. Curiously, the unsupervised node classification on the binary classification exceeds performance of supervised node classification.

**Table 1.** Results on supervised node classification.

	ORN	SRN	SRN + VIS
GNN + Majority Vote	0.580	0.588	<b>0.658</b>
Percentage Gain over ORN	-	1.4%	<b>13.5%</b>
GNN (2-class) + Majority Vote	0.905	0.905	<b>0.911</b>
Percentage Gain over ORN (2-class)	-	0.0%	<b>0.7%</b>

**Table 2.** Results on unsupervised node classification.

	ORN	SRN	SRN + VIS
GNN + Majority Vote	0.532	0.532	<b>0.534</b>
Percentage Gain over ORN	-	0.0%	<b>0.4%</b>
GNN (2-class) + Majority Vote	0.913	0.907	<b>0.915</b>
Percentage Gain over ORN (2-class)	-	-0.7%	<b>0.2%</b>

Overall, the performances on binary classification of 0.911 and 0.915 in supervised and unsupervised respectively, are comparable with the performance of RoadTagger [16], which achieved an accuracy of 93.1% on a similar dataset.

## 7. DISCUSSION & CONCLUSION

We have presented a method to enrich incomplete, structural spatial network data with complete, continuous image data. In the example of road type classification on crowd-sourced RNs, we demonstrated how low-level visual features from remote sensing data can be included into the attribute set of the RN. Moreover, we presented a segmentation based on spatial and empirical traffic information.

The results of our experiments show, that low-level visual features like pixel intensity histograms, can improve performance on both supervised and unsupervised node classification of road type labels. Moreover, we have shown that SRN+VIS can achieve performances similar to state-of-the-art models for binary road type classifications in both supervised and unsupervised settings.

Future work, should investigate to what extent the replacement of low-level visual features through CNNs like in RoadTagger [16] is beneficial for supervised and unsupervised node classifications in RN. Another interesting direction of research is to analyse how the unsupervised embeddings can be used for multiple different machine learning tasks that are relevant from a RN perspective.

## 8. REFERENCES

- [1] OpenStreetMap, "Planet dump retrieved from <https://planet.osm.org> on 2021-06-10," 2021 [Online].
- [2] Maxar Technologies, "<https://www.maxar.com/>," Aug. 2021 [Online].
- [3] The Gaia Initiative, "<https://outreach.didichuxing.com/appen-vue/>," May 2021 [Online].
- [4] William L Hamilton, Rex Ying, and Jure Leskovec, "Inductive representation learning on large graphs," in *Proceedings of the 31st NeurIPS*, 2017, pp. 1025–1035.
- [5] Stefan Funke, Robin Schirrmeister, and Sabine Storandt, "Automatic extrapolation of missing road network data in openstreetmap," in *Proceedings of the 2nd International Conference on Mining Urban Data-Volume 1392*, 2015, pp. 27–35.
- [6] William L. Hamilton, "Graph representation learning," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 14, no. 3, pp. 1–159, 2020.
- [7] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 2020.
- [8] Rafael Pires de Lima and Kurt Marfurt, "Convolutional neural network for remote-sensing scene classification: Transfer learning analysis," *Remote Sensing*, vol. 12, no. 1, pp. 86, 2020.
- [9] Rafael C Gonzalez, Richard E Woods, et al., *Digital image processing*, Prentice hall Upper Saddle River, NJ, 2002.
- [10] Zahra Gharaee, Shreyas Kowshik, Oliver Stromann, and Michael Felsberg, "Graph representation learning for road type classification," *Pattern Recognition*, p. 108174, 2021.
- [11] Tobias Skovgaard Jepsen, Christian S Jensen, and Thomas Dyhre Nielsen, "Relational fusion networks: Graph convolutional networks for road networks," *IEEE Transactions on Intelligent Transportation Systems*, 2020.
- [12] Ning Wu, Xin Wayne Zhao, Jingyuan Wang, and Dayan Pan, "Learning effective road network representation with hierarchical graph neural networks," in *Proceedings of the 26th ACM SIGKDD*, 2020, pp. 6–14.
- [13] Jielun Liu, Ghim Ping Ong, and Xiquan Chen, "Graphsage-based traffic speed forecasting for segment network with sparse data," *IEEE Transactions on Intelligent Transportation Systems*, 2020.
- [14] John E Vargas-Munoz, Shivangi Srivastava, Devis Tuia, and Alexandre X Falcao, "Openstreetmap: Challenges and opportunities in machine learning and remote sensing," *IEEE Geoscience and Remote Sensing Magazine*, vol. 9, no. 1, pp. 184–199, 2020.
- [15] Jasmeet Kaur and Jaiteg Singh, "An automated approach for quality assessment of openstreetmap data," in *2018 International Conference on Computing, Power and Communication Technologies (GUCon)*, 2018, pp. 707–712.
- [16] Songtao He, Favyen Bastani, Satvat Jagwani, Edward Park, Sofiane Abbar, Mohammad Alizadeh, Hari Balakrishnan, Sanjay Chawla, Samuel Madden, and Mohammad Amin Sadeghi, "Roadtagger: Robust road attribute inference with graph neural networks," in *Proceedings of the AAAI*, 2020, vol. 34, pp. 10965–10972.
- [17] Zhong Chen, Ting Zhang, and Chao Ouyang, "End-to-end airplane detection using transfer learning in remote sensing images," *Remote Sensing*, vol. 10, no. 1, pp. 139, 2018.
- [18] Oliver Stromann, Andrea Nascetti, Osama Yousif, and Yifang Ban, "Dimensionality reduction and feature selection for object-based land cover classification based on sentinel-1 and sentinel-2 time series using google earth engine," *Remote Sensing*, vol. 12, no. 1, pp. 76, 2020.
- [19] Andrea Tassi and Marco Vizzari, "Object-oriented lulc classification in google earth engine combining snic, glcm, and machine learning algorithms," *Remote Sensing*, vol. 12, no. 22, pp. 3776, 2020.
- [20] Natalia Verde, Ioannis P Kokkoris, Charalampos Georgiadis, Dimitris Kaimaris, Panayotis Dimopoulos, Ioannis Mitsopoulos, and Giorgos Mallinis, "National scale land cover classification for ecosystem services mapping and assessment, using multitemporal copernicus eo data and google earth engine," *Remote Sensing*, vol. 12, no. 20, pp. 3303, 2020.
- [21] Geoff Boeing, "Osmnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks," *Computers, Environment and Urban Systems*, vol. 65, pp. 126–139, 2017.
- [22] Can Yang and Gyoza Gidofalvi, "Fast map matching, an algorithm integrating hidden markov model with pre-computation," *International Journal of Geographical Information Science*, vol. 32, no. 3, pp. 547–570, 2018.
- [23] Lukas Biewald, "Experiment tracking with weights and biases," Aug. 2021 [Online], Software available from [wandb.com](https://wandb.com).