

A PATH GUIDED METHOD FOR QUICKLY LABELING ROADS IN SATELLITE IMAGES

Samet Cengiz Ozcan*, Department of Computer Engineering, Ankara Yildirim Beyazıt University, Turkey, sametc.ozcan@gmail.com

([ID](https://orcid.org/0000-0002-1825-0097) <https://orcid.org/0000-0002-1825-0097>)

Abdullah Bulbul, Department of Computer Engineering, Ankara Yildirim Beyazıt University, Turkey, abulbul@ybu.edu.tr

([ID](https://orcid.org/0000-0002-3802-1718) <https://orcid.org/0000-0002-3802-1718>)

Received: 01.04.2021, Accepted: 24.08.2021

Research Article

*Corresponding author

DOI: 10.22531/muglajsci.908214

Abstract

Determining roads from satellite images has gained more research interest after the recent progress on data-heavy machine learning methods which are also accelerated by the increased amounts of accessible data. An important challenge of learning-based approaches is obtaining labeled data to train the systems. In this study, we propose a method for quickly labeling roads over satellite images of any desired location. Our method exploits the 2D path information obtained from OpenStreetMap, an online community-contributed source of geolocated information. In this environment, roads are roughly described as line segments without their exact shapes and sizes. Using this rough information, we propose a simple interactive user interface where users easily label the road boundaries over presented satellite images. Using our approach, it is possible to rapidly label regions with different road characteristics. Such an approach allows for training separate machine learning systems for different parts of the world which would be advantageous over training a single system to identify all kinds of roads.

Keywords: City Modeling, Satellite images, OpenStreetMap, Labeling, Machine Learning

UYDU GÖRÜNTÜLERİNDE YOLLARIN HIZLI ETİKETLENMESİ İÇİN YOL KILAVUZLU BİR YÖNTEM

Özet

Uydu görüntülerinden yolların belirlenmesi konusundaki araştırmalar veri yoğunluklu makine öğrenimi yöntemlerindeki gelişmelerle birlikte hız kazanmıştır. Bu duruma erişilebilir veri miktarının artması da katkıda bulunmuştur. Öğrenmeye dayalı yaklaşımlarda karşılaşılan önemli bir zorluk, sistemleri eğitmek için gerekli etiketli verilere ulaşmaktır. Bu çalışmada, istenen herhangi bir konunun uydu görüntüleri üzerinden yolları hızlı bir şekilde etiketlemek için bir yöntem öneriyoruz. Yöntemimiz, OpenStreetMap'ten elde edilen 2B yol bilgisini kullanmaktadır. OpenStreetMap coğrafi olarak konumlandırılmış bilgiler içeren ve insanların katkısıyla oluşturulmuş çevrimiçi bir ortamdır ve burada yollar, kesin şekilleri ya da boyutları olmaksızın kabaca çizgi segmentler olarak tanımlanmıştır. Bu kaba bilgileri kullanarak, kullanıcıların sunulan uydu görüntüleri üzerinden yol sınırlarını kolayca etiketledikleri basit bir etkileşimli kullanıcı arayüzü öneriyoruz. Sunulan yaklaşım ile farklı yol özelliklerine sahip bölgeler hızlı bir şekilde etiketlenebilir. Bu durum tüm çevrelere uygun tek bir sistem eğitmek yerine daha etkili olabilecek şekilde farklı bölgelere özelleşmiş, ayrı makine öğrenimi sistemlerinin eğitilmesine de olanak tanımaktadır.

Anahtar Kelimeler: City Modeling, Satellite images, OpenStreetMap, Labeling, Machine Learning

Cite

Ozcan, S.C., Bulbul, A. (2021). "A Path Guided Method for Quickly Labeling Roads in Satellite Images", *Mugla Journal of Science and Technology*, 7(0), 19-26.

1. Introduction

Recently, 3D city modeling and simulation studies have gained more interest following the technological advances for capturing and processing data. It is possible to access 2D road networks from online sources such as OpenStreetMap (OSM). In such networks, roads are defined as polygonal path chains. This information can be used in various studies to reconstruct cities or traffic [1], [2]. One shortcoming of these sources is the lack of precise road widths. However, they are still useful as they

potentially provide prior information to the methods for automatically and precisely extracting roads from satellite images. In this study, we present a quick labeling method to feed machine learning studies on road segmentation. With the proposed study, satellite images of road segments are presented to the user such that they are always directed upwards in image coordinates and the user can easily specify the widths of presented road segments with the provided user interface. The labeled data will facilitate studies for generating precise 3D road

modeling later. After obtaining the roads, it is also possible to extract the parked or moving vehicles over the roads for an accurate traffic simulation.

We used Unity 3D game engine to present satellite images to the user in an intended manner. Linear line segments, extracted from OSM data, directs the camera to move over the roads and the system presents the roads to the user who can pick either side of a road segment before moving to another segment. Using the proposed approach, it is possible to quickly generate labeled road segments for different regions of the World. In the remaining part of the paper, first, we present the related works before explaining the details of the proposed approach. Finally, we present the results and discuss the proposed approach.

2. Related Work

Recently, road segmentation, 3D modeling, and digital twin concepts have attracted great interest. In this study, we aim to provide a labeling method for facilitating automatic segmentation of roads which forms our longer-term aim. Therefore, this section also includes road extraction and generation studies.

In one of the early studies, Bacher and Mayer [3] offer an automatic road extraction approach. The areas where water makes agriculture possible and the areas dominated by desert and dried mountain areas are modelled. The results of the study demonstrate the validity of the approach. In Kocaman et al. [4], 3D city modeling with high-resolution satellite images was studied. They show that different software and approaches are available for the creation of 3D city models. 3D city models can be created with the CyberCity ModelerTM software package. The ability to provide multi-image terrain coverage in a flight mission or satellite orbit, the multi-image matching approach, has been reported to cause reduced problems due to occlusions, multiple solutions, surface discontinuities, and higher measurement accuracy. Y. Wang et al. [5] use urban roads. This article focuses on the automatic extraction of ways in urban areas from high-resolution satellite imagery.

For road detection, several methods have been developed in recent years. Recent improvements on road extraction studies mostly utilize deep learning and neural networks based methods. Deepan et al. [6] investigated the process of dividing the RGB image into four blocks of 250x250 pixels to extract roads. In other similar studies, Xu et al. [7] and Zhang et al. [8] used incremental learning, machine learning and deep learning for road detection. Taşpınar [9] used CNN in his study. He also used R-CNN, Fast R-CNN and Faster R-CNN neural networks. These neural networks detected the presence of objects in the images and returned the bounding boxes of the object. There are neural networks that can perform segmentation tasks, which determine which object the pixel in the image belongs to. Wang et al. [10] studied and examined the use of deep convolutional neural networks (DCNN) for road

extraction. Also Xia et al. [11] show that their proposed solution successfully identifies road networks from complex situations with an overall accuracy of more than 80 percent in distinguishable areas. Wang et al. [12] focuses on automatic way extraction in urban areas from high-resolution satellite imagery. It focuses on features that reflect the proportion of bright areas, the directional consistency of the edges, and the road characteristics of local binary patterns. AdaBoost is adopted to select the most effective features from these features. Paths are detected with a sliding window with learning results and the path link is validated by merging. Experimental results on real Quickbird images show the effectiveness and robustness of the method used.

Lane detection is important for the development of smart vehicles and several recent studies focus on lane level road extraction. A lane detection algorithm has been proposed by Cao and Song et al. for intelligent vehicles in complex road conditions and dynamic environments. [13]. Besides, Malladi used object detection in his study to detect objects in satellite imagery with supervised and unsupervised learning methods [14]. Dai et al. [15] propose the method of high-resolution optical satellite image strip levelling for road extraction. The proposed method not only ensures the accuracy of the lane level guidance but also greatly increases the automation of the guidance. Azimi and Fischer [16] proposed a study where elaboration of public transport is based on the creation of smooth road maps and lane-based models. It contributed to overcoming the difficulties associated with the semantic strip marking segmentation of aerial imagery. The first high-quality data set used in experiments involving a wide range of situations and lane marking classes representing today's transport systems is presented.

We propose using OpenStreetMap data to label roads segments and simplify road extraction problem. Similarly, there are studies in the literature that employ additional data sources for various purposes about road extraction. To achieve their goals, an addressing system is provided by Mattyus et al. [17] to generate addresses compatible with the routing topology. They provided an approach for developing world maps with aerial imagery. Contextual models utilizing maps are very effective for many recognition and localization tasks. It is recommended to use aerial images to develop existing world maps. To achieve that OpenStreetMap is used, which is parameterized in terms of the location and width of the path segment centerlines, and formulated as an inference problem in random space. On the other hand, there is a productive system [18] that can be operated to any planned or unpaired location region that produces a street labeling solution. An algorithm is created that generates meaningful addresses for locations that do not have street names or addresses and do not match. An automated algorithm is provided to perform this task using machine learning and computer vision approaches from satellite images. Also in a study to find solutions to the problem of city travel with

artificial intelligence, OpenStreetMap was used as a source and information and results about the map were examined [19].

3. Overview

We are proposing a quick road labeling method as part of a larger aim of auto-extraction of roads from satellite images. The overview of the study is shown in Figure 1.

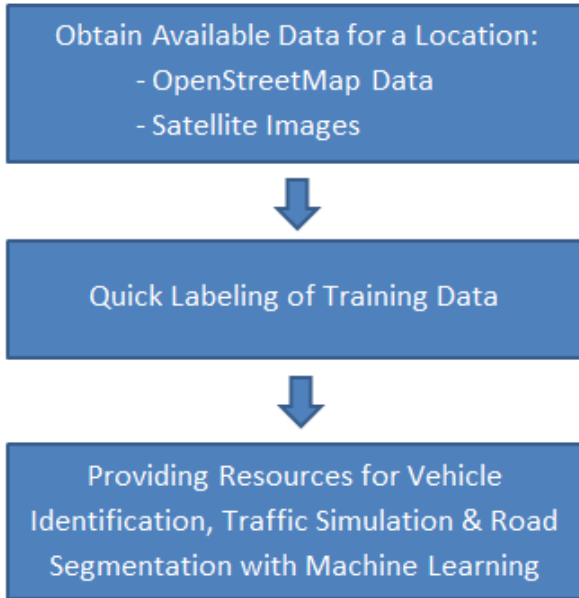


Figure 1. Overview of the in this study.

In this study, given the geo-coordinates of a region, available data is acquired from online available sources as the initial step. This data includes OpenStreetMap data: providing textual descriptions of 2D plans of buildings, roads, and additional information; and satellite images of the same region which is acquired from Google Maps in our case. Using the available sources we provide a quick user interface for the users to specify borders of road segments. The acquired information is kept to be used as labeled data to train neural networks in the later steps.

Later steps, which are currently ongoing and have not been presented here include using machine learning with the labeled data for automatic extraction of the roads from satellite images. After segmenting out the roads in a detailed manner, this information can be used to generate detailed 3D roads by also employing the elevation information. The last step is to use the generated roads for traffic simulation or other similar applications such as games. An additional planned study is detecting the vehicles in the extracted road segments and utilizing that information in the simulation application.

4. Obtaining Available Data

In the proposed study, we fundamentally need two kinds of information. The first one is a rough estimate of 2D road paths. The second one is satellite images of the same

regions. Also, it should be possible to align and overlay the two types of sources.

OpenStreetMap [20] is used to obtain the first kind of information. A sample image from OSM is shown in Figure 2. OSM provides roughly accurate 2D city data, where each building and other closed areas are expressed with polygons or polylines. It is also possible to access geographic coordinates for each corner of these polygons. To obtain a 2-dimensional plan of the area to be modelled, data is downloaded from the OSM, where each building, road and similar area are created by users in the form of 2-dimensional polygons and lines. This data consists of an XML file with the extension ".osm". An example field definition is shown in Figure 3. In the example, the way includes the polygon that forms the 2D boundaries of the relevant building and optional information such as name, address, number of floors. Each vertex of the polygon consists of a node "nd" and these nodes contain coordinate information. 2D polygons created by reading the ".osm" file containing the modelled region can be raised from the ground and converted into simple 3D models [21]. Similar to buildings, roads are expressed as "way"s that contain "nd" references as control points (or corners). As shown in Figure 3, a way definition optionally includes additional information such as "highway", "secondary", "oneway" etc. Although this additional information provides useful features, they are not sufficient by themselves to create road models with accurate shapes and widths for purposes such as simulation or reconstruction.

Using information from OSM, it is possible to create 2D roads as shown in Figure 4. Unity 3D Engine was used to create and visualize the road models.

The second type of information is obtained by Google Maps API. It provides the satellite images of a desired location as shown in Figure 5. Since it is not possible to obtain a single high quality image of a large area, we first divide a specified region into grids and combine the satellite images of each grid to generate the whole area as applied in [21]. After the city was built on Unity, it resulted as shown in Figure 6. Sample city models shown over a satellite image illustrates the alignment of the two types of localized sources.

In practice, a city is created in Unity using OSM data. In the created city, the user only sees the satellite images; therefore, all other details are hidden. Afterwards, to improve the city's roads, the user is asked to select the paths from the images to tag them. The selected paths are saved and labeled data is provided for machine learning to be trained later.



Figure 2. OpenStreetMap view of Ankara Yıldırım Beyazıt University

```

.....
</way>
<way id="32802166" visible="true" version="5" changeset="16631617"
timestamp="2013-06-20T15:37:43Z" user="SathyaV" uid="1233272">
<nd ref="2343952409"/>
<nd ref="2343952407"/>
<nd ref="2343952405"/>
<nd ref="2343952404"/>
<nd ref="2343952402"/>
<nd ref="2343952401"/>
<nd ref="2343952398"/>
<nd ref="2343952395"/>
<nd ref="2343952386"/>
<nd ref="2343952374"/>
<nd ref="369639579"/>
<nd ref="369639580"/>
<nd ref="369639581"/>
<nd ref="2343947663"/>
<tag k="highway" v="secondary"/>
<tag k="oneway" v="yes"/>
</way>
<way id="32921902" visible="true" version="7" changeset="43882296"
timestamp="2016-11-22T22:19:36Z" user="Muhammed Zayif" uid="4843145">
.....
    
```

Figure 3. An example from the osm file



Figure 4. Unity real city roads

5. Quick Labeling Method

Several steps are performed for the Quick Labeling Method. A summary of these steps are presented below.

- The user indicates geo-coordinates of the area of interest.
- Required data for the specified region is obtained from OSM as an ".osm" file.
- Satellite images of the same region are downloaded from Google Maps through its Static Maps API.
- 2D road paths and satellite images are loaded and aligned using geo-coordinates.
- A camera scans the satellite image by following the road segments indicated in the osm file.

- The camera presents a vertically aligned top view of each road segment to the user.
- The user picks two ends of the road to specify its position and width. If there are multiple road segments due to traffic islands, the user can label them separately.
- Satellite images and their labels are saved for the region of interest to form the machine learning dataset.

These steps are sequentially followed in our labeling management. All these steps are detailed in the following paragraphs.



Figure 5. Google Satellite image example



Figure 6. Sample view from generated city model.

During the creation of the virtual city we use Unity Game engine. It is important to provide high resolution satellite images to the user to get accurately labeled roads. The satellite images are obtained from Google Maps API where each image has a resolution limit of 640x640 pixels. Thus, we need to download multiple satellite images of the region and combine them to get sufficiently high quality images as applied in [21]. OSM related road geometry is rendered onto Google's satellite images. Having an accurate alignment of those two sources are crucial for the proposed method. Figure 7 visualizes the alignment of the roads and satellite images. As seen from

this figure, the two sources are sufficiently well aligned while only for a minority of regions there are misalignments. In our experiment, we observe that the road segment is close to the center of the present satellite image as expected (See Figure 12).

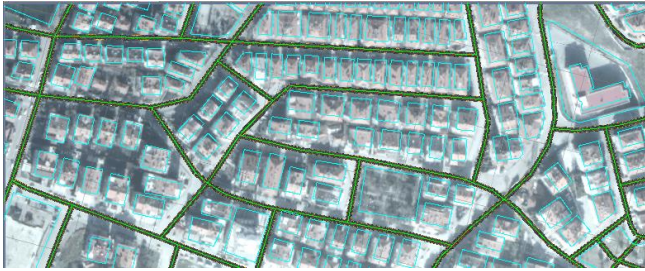


Figure 7. Alignment of the OSM Roads and Google Maps Satellite images

While the city is being loaded, the Camera is focused on the roads on the map. The camera has a bird's-eye view on the map and it is positioned directly over the center of the current road segment so that the road view is centered on the screen. In our study, roads have always been considered as linear line segments. Bezier curves or b-spline algorithms can be applied in case of problems for curvy sections of the road.

In addition to placing the camera over the road segments, we need to adjust its orientation. By orienting the camera such that its up direction follows the road segment as illustrated in Figure 8, we provide user a vertical image of the road. By reorienting the satellite images, the road segmentation problem is reduced to find the width of the road.

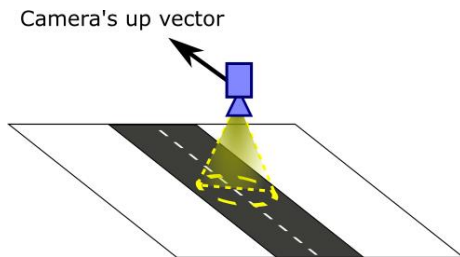


Figure 8. Camera placement over a road segment.

For labeling, the user is asked to select the path with the mouse. A line is drawn between the two selected points as show in Figure 10. The user uses the left / right arrow keys to move on the map. In order not to have the same views that are not very different from the previous labeled view, the user can skip a number of road segments between each two labeling. It is possible to specify the number of road segments to be skipped at each step. The user can also skip labeling if the current road segments are not suitable for labeling.

In some cases, the user cannot label the entire road by drawing a single line since the roads include traffic islands etc. In such cases, multiple selections can be made. In addition, it is possible to have a big variety on road widths resulting in the road to cover very small part

of the screen in some cases or the road to exceed the screen boundaries in some others. To facilitate this issue, the roads can be labeled at different zoom levels with the zoom in-out feature. Zoom level can be controlled with the mouse scroll wheel or the keyboard. When saving the label data zoom level is also recorded.

When labeled and switched to the other view, the direct saving of the tagged image and label data occurs. The coordinates of the selected path are saved in the screenshot and text file with the same unique way id name. In the text file, in addition to the x_1 , x_2 coordinate the bounding box labels x_1 , y_1 , x_2 and y_2 are saved as two separate outputs with their images to be used when needed. Here, as seen in Figure 13, Where x_1 and x_2 are the selected line coordinates, y_1 is assigned the upper left corner 0 value and y_2 is assigned the lower right 1 value. It is also saved in separate folders according to the zoom in-out levels. If there is more than one label in the view, an image and label data in more than one line in a text file of this image are saved.

6. Result & Discussion

In this study, we have presented our quick road labeling method for a given geo-location. In our experiments, Ankara's Etlik, Madrid's Sol etc. district is integrated as a virtual city and we have performed road labeling on this region. The results are as follows. The roads were pre-drawn with texture. The screen for the user to choose is in Figure 9. Here comes the unselected and non-textured roads.



Figure 9. Road non-textured

In Figure 10, you can see the line drawn after selecting it. To improve the paths, the later path textures were made invisible from Figure 11. When the user labels and saves, the line is drawn on the screen disappears and becomes the same as in Figure 9, so the screenshot is saved. Various examples of labeled roads are available in Figure 12.

Users choose the path on this screen. If necessary, reset and the selected image with the new selection. And the following results are obtained;

1. Coordinates with the name way id are saved as a text file.
2. It was saved with the same name as the screenshot in the image.



Figure 10. Path selection and line for output



Figure 11. Road texture drawn



Figure 12. Labeled examples of roads in Ankara's Etlík

```
File name: 32802165_3 (way id)

First label(x1,x2):
0.3199528 0.4840614

Second label(x1,x2, y1 and y2)
0.3199528 0.0 0.4840614 1.0
```

Figure 13. Content of the saved file for road border specification

The screen image of the road without the tags on it and the coordinates as follows will be taken as output. It is shown in Figure 13. As explained and detailed in step 8 in the V. Method section, the file name is given according to the way id and node. Only selected points x_1 , x_2 in the first label. On the second label, y_1 is assigned the upper left corner 0 and y_2 is assigned the lower right 1 value, and the x_1 , x_2 , y_1 and y_2 points are specified. These outputs can be shaped according to the machine learning algorithm to be used.

With the proposed approach, a user can quickly specify road borders. In order to test the efficiency of the proposed labeling method, we have observed four subjects while they perform labeling. According to our observations, it takes 10-15 minutes for an average user to specify the borders of 100 road segments. Also, the subjects choose to label 74\% of the presented road segments while skipping the rest of them. The reasons for skipping several cases include not having well-defined borders of the road segment, e.g., a junction; invisible road segments due to occlusion and shadows from tall buildings or trees; or misalignment of the two sources of data (OSM and Google Maps). In general, the provided method allows users to label road segments in a simple and efficient manner.

The broad goal of our study is to create the roads of any city. To achieve that, using the proposed labeling method, users can label a small portion of the road segments by defining their widths. This labeled data can be used by machine learning methods to automatically extract the rest of the road segments. This approach enables separately training road segmentation methods for cities with different visual characteristics.

The main purpose of the proposed method is to provide an easy to use road labelling system for satellite images. For this purpose, our labeling system can be used anywhere. Our labeling system can be used wherever we want, such as Ankara's Etlik in Figure 12 and Madrid's Sol in Figure 14.



Figure 14. Labeled examples of roads in Madrid's Sol

7. Conclusion & Future Work

This study was used to select paths and find widths for data provided for the 3D road model. The project aims to model the path with road width data on the map. To do this, it is necessary to create a data set and learn a machine with these data. As a result, data is provided for intelligent city simulation and roads in the city are improved. The aim is to model the roads in the city close to the actual width as in Figure 16.

In the future, many different adaptations, tests and experiments will be carried out and the application will continue to be developed. A deeper analysis will be made and more meaningful data will be provided to the machine learning work. Using the same application on different roads and extracting appropriate data to find the correct path widths as shown in the Figure 16 as a result of machine learning.

One advantage of the proposed quick labeling method is that it enables rapidly forming a training dataset for any desired region. Instead of using a single network that works everywhere, we provide an environment to feed networks which can be specifically trained for the region of interest so that it has the potential to adapt local features and varieties of that region. These locations can be operated whether on complex roads, uncomplicated roads, or places with very busy roads. By Google 2D

satellite images, variety of the roads can be exemplified as follows [22]: Figure 15, In Turkey, for example, to place the complex ways in Ankara, Bağlıca neighborhood. The other image in Figure 15 Turkey, Istanbul neighborhood at night Hadımköy put neighborhoods and sample locations with complex paths. The final image in Figure 15 shows an example from Madrid, Spain, with no less complicated and very busy roads. As seen, we can apply the proposed method in any desired place having its own features.



Figure 15. Google Satellite image samples. From top to bottom: Bağlıca, Ankara; Hadımköy, Istanbul; and Madrid.



Figure 16. Road building in the city [23].

8. References

- [1] Yildirim, O., Cetin, M., Erdogan, M., Gurleyuk, N. and Bulbul, A. "Race on your street," in *Eurasia Graphics 2018*, 2018.
- [2] Bulbul, A. and Dahyot, R. "Populating virtual cities using social media," *Computer Animation and Virtual Worlds*, vol. 28, no. 5, p. e1742, 2017.
- [3] Bacher, U. and Mayer, H. "Automatic road extraction from irs satellite images in agricultural and desert areas," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 35, p. B3, 2004.
- [4] Kocaman, S., Zhang, L., Gruen, A. and Poli, D. "3d city modeling from high-resolution satellite images," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, no. 1/W41, 2006.
- [5] Wang, Y., Tian, Y., Tai, X. and Shu, L. "Extraction of main urban roads from high resolution satellite images by machine learning," *Asian Conference on Computer Vision*. Springer, 2006, pp. 236–245.
- [6] Deepan, P., Abinaya, S., Haritha, G. and Iswarya, V. "Road recognition from remote sensing imagery using machine learning," *International Research Journal of Engineering and Technology*, vol. 5, no. 3, pp.3677–3683, 2018.
- [7] Xu, Y., Xie, Z., Feng, Y. and Chen, Z. "Road extraction from high-resolution remote sensing imagery using deep learning," *Remote Sensing*, vol. 10, no. 9, p. 1461, 2018.
- [8] Zhang, J., Chen, L., Wang, C., Zhuo, L., Tian, Q. and Liang, X. "Road recognition from remote sensing imagery using incremental learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 11, pp. 2993–3005, 2017.
- [9] Taspınar, A. "Using convolutional neural networks to detect features in satellite images," <https://ataspinar.com>, 2017.
- [10] Wang, J., Song, J., Chen, M. and Yang, Z. "Road network extraction: A neural-dynamic framework based on deep learning and a finite state machine," *International Journal of Remote Sensing*, vol. 36, no. 12, pp.3144–3169, 2015.
- [11] Xia, W., Zhang, Y.-Z., Liu, J., Luo, L. and Yang, K. "Road extraction from high resolution image with deep convolution network—a case study of gf-2 image," *Multidisciplinary Digital Publishing Institute Proceedings*, vol. 2, no. 7, p. 325, 2018.
- [12] Wang, W., Yang, N., Zhang, Y., Wang, F., Cao, T. and Eklund, P. "A review of road extraction from remote sensing images," *Journal of traffic and transportation engineering (english edition)*, vol. 3, no. 3, pp. 271–282, 2016.
- [13] Cao, J., Song, C., Song, S., Xiao, F. and Peng, S. "Lane detection algorithm for intelligent vehicles in complex road conditions and dynamic environments," *Sensors*, vol. 19, no. 14, p. 3166, 2019.
- [14] Malladi, C. "Detection of objects in satellite images using supervised and unsupervised learning methods," *Faculty of Computing Blekinge Institute of Technology*, 2017.
- [15] Dai, J., Zhu, T., Zhang, Y., Ma, R. and Li, W. "Lane-level road extraction from high-resolution optical satellite images," *Remote Sensing*, vol. 11, no. 22, p. 2672, 2019.
- [16] Azimi, S. M., Fischer, P., Körner, M. and Reinartz, P. "Aerial lanenet: Lane-marking semantic segmentation in aerial imagery using wavelet-enhanced cost-sensitive symmetric fully convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 5, pp. 2920–2938, 2018.
- [17] Mattyus, G., Wang, S., Fidler, S. and Urtasun, R. "Enhancing road maps by parsing aerial images around the world," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1689–1697, 2015.
- [18] Demir, I., Hughes, F., Raj, A., Dhruv, K., Muddala, S. M., Garg, S., Doo, B. and Raskar, R. "Generative street addresses from satellite imagery," *ISPRS International Journal of Geo-Information*, vol. 7, no. 3, p. 84, 2018.
- [19] Ozcan, S. C. and Kaya, H. "An analysis of travelling salesman problem utilizing hill climbing algorithm for a smart city touristic search on openstreetmap (osm)," in *2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*. IEEE, pp. 1–5, 2018.
- [20] Wiki openstreetmap, "About openstreetmap," <https://wiki.openstreetmap.org>, 2020.
- [21] Bulbul, A. "3d city modeling from accessible online sources," *8. Ulusal Savunma Uygulamaları Modelleme ve Simülasyon Konferansı (USMOS)*, pp. 315–326, 2019.
- [22] Google Earth earth.google, "Create stories and maps," <https://earth.google.com/web/search>, 2021.
- [23] Made in Bocholt madeinbocholt, "Virtuelles Stadtmodell auf Basis von Geodaten," <https://madeinbocholt.de/virtuelles-stadtmodell-auf-basis-von-geodaten>, 2021.