ARTICLE

Estimating traffic in urban areas from very-high resolution aerial images

Jarle Hamar Reksten and Arnt-Børre Salberg

Norwegian Computing Center, P.O. Box 114 Blindern, N-0314 OSLO

ARTICLE HISTORY
Compiled March 10, 2021

ABSTRACT
Traffic estimation from very-high-resolution remote sensing imagery has received increasing interest during the last few years. In this article, we propose an automatic system for estimation of the annual average daily traffic (AADT) using very-high-resolution optical remote sensing imagery of urban areas in combination with high-quality, but very spatially limited, ground based measurements. The main part of the system is the vehicle detection, which is based on the deep learning object detection architecture mask region-based convolutional neural network (Mask R-CNN), modified with an image normalization strategy to make it more robust for test images of various conditions and the use of a precise road mask to assist the filtering of driving vehicles from parked ones. Furthermore, to include the high-quality ground based measurements and to make the traffic estimates more consistent across neighboring road links, we propose a graph smoothing strategy that utilize the road network. The fully automatic processing chain has been validated on a set of aerial images covering the city of Narvik, Norway. The precision and recall rate of detecting driving vehicles was 0.74 and 0.66, respectively, and the AADT was estimated with a root mean squared error (RMSE) of 2279 and bias of -383. We conclude that separating driving vehicles from parked ones may be challenging if vehicles are parked along the roads, and that for urban environment with short road links several remote sensing images covering the road links at different time instances are necessary in order to benefit from the remote sensing images.

KEYWORDS
vehicle detection; traffic estimation; deep learning; graph smoothing

1. Introduction

Road traffic density is a crucial parameter for road applications, including planning, construction and operation of road networks, but also to estimate the level of pollution caused by traffic. In Norway, traffic density is currently regularly measured at 9300 ground-based counting stations throughout the Norwegian road network, but this is far from satisfactory when it comes to geographical coverage. Mobile equipment such as radar is also applied, but those counts are limited in both space and time, the mobile device is expensive, and operation is time-consuming. It is also challenging to distinguish vehicle weight classes in data from mobile radar devices.

Due to these limitations on equipment, operational costs, geographical coverage, and resolution with respect to vehicle weight classes, remote sensing images are now being evaluated for estimating traffic Larsen, Haug, and Aldrin (2008); Eikvil, Aurdal, and Koren (2009); Larsen, Salberg, and Eikvil (2013); Leitloff et al. (2014); Audebert, Le Saux, and Lefèvre (2017). Very-high-resolution (VHR) aerial images

CONTACT J. H. Reksten. Email: jarle.reksten@nr.no

allow for automated monitoring of road traffic situations Hinz and Stilla (2005); Zhao and Nevatia (2003); Eikvil, Aurdal, and Koren (2009); Leitloff, Hinz, and Stilla (2010); Larsen, Salberg, and Eikvil (2013); Leitloff et al. (2014); Liu and Mattyus (2015); Paisitkriangkrai et al. (2015); Kampffmeyer, Salberg, and Jenssen (2016); Audebert, Le Saux, and Lefèvre (2017). Using remote sensing imagery, reduced costs and immense improvements in geographical coverage are obtained compared with the use of mobile equipment. These benefits are also observed in other applications (see e.g. Lato, Frauenfelder, and Bühler (2012); Lv et al. (2019); Yu et al. (2018)).

To assess the traffic density in a road network, a commonly used measure is the annual average daily traffic (AADT), which is the average number of vehicles passing a given point each day. Larsen, Haug, and Aldrin (2008) showed that the AADT of a given road link can be estimated from vehicle counts obtained from road traffic snapshots (e.g. VHR aerial or satellite images) if the length of the road link is sufficiently long, or a sufficient number of traffic snapshots is available. The conversion from road traffic snapshot vehicle counts to AADT also requires that the vehicles are driving at a known speed, e.g. the speed limit.

Several algorithms for vehicle detection in remote-sensing data have been developed during the last decades. Eikvil, Aurdal, and Koren (2009) proposed the use of image segmentation with global thresholding followed by two stages of object classification. Their study site was located at the inner city roads in Oslo, Norway. The results showed that the segmentation step was the limiting factor of the proposed method, with approximately 20% of the vehicles were lost in this process due to low contrast and fragmented objects. An adaptive boosting (AdaBoost) classifier in combination with Haar-like features were applied by Leitloff, Hinz, and Stilla (2010) to detect single vehicles for VHR optical satellite images. Moreover, vehicle queues were detected using a line extraction technique, and single vehicles were determined within the queue using a robust parameter estimation. The authors reported that a detection rate of over 80% was possible. The advantage of the scheme proposed by Leitloff et al. is the potential to detect and count vehicles in queues. However, some limitations were reported related to missing detection of vehicles with low contrast to the surrounding and false detections caused by shadows casts. Larsen, Salberg, and Eikvil (2013) proposed a system for automatic traffic monitoring using very-high resolution satellite imagery. Central in the proposed system was a segmentation strategy, where interesting features (blobs) in the image were located using a scale space-filtering approach. From each detected blob, spectral, geometric, and context-based features were extracted and classified into six vehicle classes and a non-vehicle class. An advantage of the proposed system was that it included a fully automatic approach from cloud detection to vehicle counting, however, a major limitation of the system is that it was designed for rural conditions. Leitloff et al. (2014) used a three stage procedure for detecting vehicles: first an AdaBoost classifier with Haar-like features, then blob detection for reducing the number of vehicles hypothesis, and finally a support vector machine (SVM) on various geometric and radiometric features. They also used a road database as a prior to detect only along the roads in a certain direction. The performance of the system by Leitloff et al. (2014) was in general very good. Some false positive detections was observed due to the sensitivty of the Haar-like features for strong edges. However, their system was not designed for traffic monitoring during disaster and mass events, and utilizes optical image sequences that are processed real time on board aircraft to estimate road traffic information.

After the introduction of deep learning to the computer vision community Krizhevsky, Sutskever, and Hinton (2012); Zhao et al. (2019), several studies have applied deep learning to detect vehicles in VHR optical images (see e.g. Paisitkriangkrai et al. (2015); Kampffmeyer, Salberg, and Jenssen (2016); Audebert, Le Saux, and Lefèvre (2017); Sabour, Frosst, and Hinton (2017); Yu et al. (2019)). Previous work that has focused on semantic segmentation of urban environments (including vehicles) includes among others Paisitkriangkrai et al. (2015), who proposed a scheme for semantic segmentation using a combination of a patch-based convolutional neural network (CNN) and a random forest classifier that is trained on hand-crafted features. To increase the classification accuracy further, a conditional random field (CRF) was used to smooth the final pixel labeling results. Inspired by the fully convolutional network (FCN) architecture Long, Shelhamer, and Darrell (2015), Kampffmeyer, Salberg, and Jenssen (2016) designed an architecture that allowed end-to-end learning of pixel-to-pixel segmentation of VHR optical remote sensing images. Training of the FCN network was done by weighting each class with its median frequency to account for class imbalance, i.e. some classes, like the vehicle class, contains much fewer pixels than other classes. This enhanced the segmentation performance of the vehicle class significantly. Even though the focus of Kampffmeyer, Salberg, and Jenssen (2016) was semantic segmentation, their approach also demonstrated the potential of deep neural networks to detect vehicles in VHR optical remote sensing images. Audebert, Le Saux, and Lefèvre (2017) proposed a three-step segment-before-detect pipeline to perform vehicle extraction and classification in VHR remote sensing data over urban areas. The three steps consisted of: (i) semantic segmentation using a FCN, (ii) vehicle detection by bounding box regression, and (iii) object-level classification with a traditional CNN. The results obtained by Audebert, Le Saux, and Lefèvre (2017) were in general good. However, the semantic maps created by SegNet can be noisy, with blurred transitions between the classes. To compensate for this, Audebert, Le Saux, and Lefèvre (2017) applied several morphological operations. Recently, novel deep learning image analysis architecture such as capsule networks Sabour, Frosst, and Hinton (2017) has also been applied for vehicle detection in VHR remote sensing images Yu et al. (2019). Their approach was based on a superpixel-based patch generation strategy and the convolutional capsule network. Capsule networks can effectively handle objects of varying sizes and orientations, and therefore provide a good match to the task of detecting vehicles in remote sensing images. The proposed method performed effectively in handling various vehicle conditions and achieved promising vehicle detection results.

In this article, the overall goal is to develop methodology for automatic estimation traffic using VHR remote sensing images. In order to fulfil this goal, key components to be developed are an automated approach for robust detection of vehicles in VHR remote sensing data, utilize the existing road network for improved traffic estimating, and enable calibration towards existing high-quality traffic estimates. Evidently, the ideal choice of methods for image analysis-based vehicle detection depends on the conditions in the image, which again depends on location, type of road, traffic density, etc.

The proposed system to extract vehicle counts from VHR remote sensing images consists of a fully automatic processing chain, where the only input required is the remote sensing image and a road vector file that identifies the geographic coordinates of each road segment, and a high-quality road mask (Figure 2 (*b*)). The system includes a new normalization approach for improved generalization of the neural network model to unseen scenery. In addition, a novel graph theory approach for denoising and automatic conversion from the neural network output counts into annual average daily traffic (AADT) is described. The vision is to apply the traffic estimates provided by the proposed system for each road link to improve local air quality estimates using a model based on the European monitoring and evaluation programme (EMEP), referred to as the urban EMEP (uEMEP) model. uEMEP is a methodology for subgrid downscaling of EMEP gridded consentrations, which are about 7×7 km^2, to say 50×50 m^2 Wind and Denby (2017).

2. Data

In this paper, we develop the deep learning algorithm using the public available international society for photogrammetry and remote sensing (ISPRS) two-dimensional (2D) semantic labeling contest datasets Int. Soc. Photogramm. Remote Sensing (ISPRS) (2018). The ISPRS datasets are comprised of aerial images over two cities in Germany: Potsdam and Vaihingen, which have been labelled with six of the most common land cover classes: impervious surfaces, buildings, low vegetation, trees, cars and clutter.

2.1. *Training data*

For training the vehicle detection module, both the ISPRS Potsdam and ISPRS Vaihingen datasets were utilized. Both datasets contain several patches of true orthophoto aerial imagery in addition to a ground truth labelling for part of the data. The ISPRS Potsdam dataset contains 38 patches of 6000×6000 pixels with 5 cm resolution and red green blue (RGB) channels. Of the 38 image patches, 24 of them includes a ground truth labelling. The ISPRS Vaihingen dataset contains 33 patches of different sizes with 9 cm resolution and infrared red green (IRRG) channels. Of the 33 patches, 16 includes a ground truth labelling.

2.2. *Road database*

In this work, all the detected vehicles are assigned to a unique road link object from "Norsk Vegdatabase (NVDB)" (the norwegian road database). In addition to the geographic location of the road links, the NVDB has also been utilized to extract information about the speed limit. For most road links, the speed limit was included in the database, however, for a few instances this was not the case. For those road links, the standard speed limits were used. In Norway these are 50 km h^{-1} in residential areas, and 80 km h^{-1} in rural area areas. The NVDB does not contain any information about parking regulations associated to each road segment.

The NVDB is far from accurate in terms of the geographic location of the road segments. In some cases, the center line is even located outside of the road. However, in Norway there exist a database, "Felles Kartdatabase - Veg (FKB-veg)" (common map database - road), which includes polygons that accurately places the edge of the road. Please note that the FKB-veg database does not include the road link objects or any references to those objects.

Some of the road links in the NVDB database includes estimated AADT values. These have been estimated from ground based counting devices, and are regarded to be of high quality.

2.3. *Test data*

The test data consist of aerial images of the city Narvik in Norway, one set from 2012 and one from 2017. The resolution of the images is 10 cm, and the images contain the three RGB channels. However, to fit the training data, we only consider the RG channels (see Section 3.1).

In order to be able to get some statistics from the vehicle detection step, a manual inspection of the 2012 dataset was performed in order to identify the vehicles. In total 1707 vehicles were found, and of them only 123 were labelled as driving vehicles, whereas the remaining 1584 vehicles were assumed to be parked.

3. Vehicle detection

3.1. *Data preprocessing*

The training datasets presented in Section 2.1 and the test data in Section 2.3 have only the red and green (RG) channels in common. Both the Potsdam dataset and the test dataset have the standard RGB channels, whereas the Vaihingen imagery has infrared (IR), red and green channels. Hence, in order to use both the training datasets, only the the red and green channels were used for training the vehicle detection model. Preliminary experiments showed that the performance of the detection model were better when including the Vaihingen dataset and using only two channels compared to using only the Potsdam dataset for training. Since pre-trained models which expects three channels were used, the green channel were also used as a blue channel, thereby using images with red, green and green instead of red, green and blue channels respectively.

In order to reduce the effect from different sensors, different weather conditions, etc., a normalization of the images was performed. However, a standard normalization with regard to mean and standard deviation calculated with the entire image as basis is highly dependent on the scene. I.e., an image covering a downtown area would be normalized differently compared to an image from a rural area where more green surroundings will impact the normalization. Since vehicles in most cases are located on impervious surfaces, using a road mask to extract only pixels from impervious surfaces, will yield a more consistent normalization. Preliminary experiments showed an improved stability from the different images compared to a standard normalization .

Finally, the resolution of the training data was changed in order to match the input data, in addition the training images were cropped into smaller image patches more suitable for training of a neural network.

The image normalization scheme was also conducted on the test images. However, in contrast to the training images that includes fully segmented labelling where the impervious surfaces are denoted as a separate class, such labels are not available for the test images. Fortunately, the road data base FKB-veg includes an accurate delineation of the roads, and were thus used for masking the pixels covering the roads. For areas which are not covered by the FKB-veg, it is also possible to derive a road mask using Open Street Map or other similar databases. Although, the neural network does not require a certain size, the test data images are split into smaller tiles in order to avoid too high memory usage.

3.2. *Instance segmentation and vehicle counting*

In order to detect and count the individual vehicle instances in the images, we trained the object detection and segmentation deep learning network mask region-based convolutional neural network (Mask R-CNN) He et al. (2018) with a FPN-101 backbone. A PyTorch implementation of the Mask R-CNN network was downloaded from Massa and Girshick (2018). In addition pre-trained weights were downloaded and utilized for the training.

During the training of the network, the training data was augmented by doing random flipping, rotation and crop of the image patches. In addition various random noise were introduced in order to improve the generalization properties of the model.

The output of the Mask R-CNN network is a binary mask for every detected vehicle instance. From each such binary mask a polygon describing the boundary is extracted. By representing each vehicle with the centroid of the polygon, every detected vehicle is mapped to the nearest road segment and counted.

3.3. *Parked vehicles and other false detections*

Since the aim is to estimate air pollution from traffic, parked vehicles are not to be included in the vehicle count, and are thus regarded as false detections. In downtown or residential areas where it is allowed to park along the side of the road, these false detections are significant, and hence it is desirable to remove these detections.

However, since parked vehicles, especially when they are parked along the roadside, look very similar to driving vehicles in a still image, these false detections have proven difficult to eliminate. In fact, in some cases it is difficult also for humans to distinguish if a vehicle is parked or driving. However, in most cases humans are able to use knowledge about traffic rules, cultural habits, local information and so on in order to accurately determine the parked/driving state of a vehicle. Unfortunately, this is not easily converted into features applicable for a machine learning algorithm. Currently, three schemes have been adopted in order to remove false detections:

- Firstly, we remove detected vehicles that are located above a maximum distance from the center of the road link. This effectively removes detections on e.g. nearby parking lots or driveways. By setting a strict maximum distance, for instance 40% of the road width, it is also possible to remove detections located on the side of the road. However, since the georeference of the road database is imprecise, this may remove correctly detected vehicles.
- Secondly, in order to remove vehicles which are parked along the side of the road, we utilize the FKB-veg database that includes exact location of the edge of the roads. By shrinking these polygons by 50 cm and remove all vehicles intersecting the shrunken polygons, a significant ratio of the parked vehicles are removed from the traffic count.
- The third scheme for removing parked vehicles, is to estimate the direction of the detected vehicle, and calculate the angle between the vehicle and the nearest part of the road link. A vehicle moving along the road, is expected to have an angle difference of about 0° or 180°, and thus it is possible to impose a restriction on this angle difference, and thereby remove vehicles far from parallel to the assigned road link. This is an effective way of removing parked vehicles that are more perpendicular to the road.

4. Traffic estimation

4.1. *From vehicle counts to traffic*

The vehicle detection and mapping produces a count of vehicle for each road segment in the road database. An instantaneous traffic estimate for road segment i may then be obtained by:

$$T_i = N_i \times \left(\frac{v_i}{l_i} \right), \tag{1}$$

where N_i is the number of vehicles mapped to the road segment, v_i is the speed of the vehicle(s) and l_i is the length of the road segment. Both the number of vehicles N_i and the road segment length l_i is known as the latter is easily calculated from the road segment object from the road database. Unfortunately, the speed of the vehicles is difficult to estimate based on a single still image. To obtain a speed estimate based on still imagery, a time series of at least two images in a reasonable temporal distance is required. Fortunately, the road database contain the speed limit of most of the road segments. Using the speed limit as estimate for the vehicle speed is far from perfect, but on the main roads this is believed to be a reasonable estimate as long as the images are not obtained during the rush hours, which is true for the test images used in this work.

Another problem is connected to the length of each road segment: a significant portion of the road segments are relatively short, only a few metres in some cases. For such very short road segments the traffic estimate will become either very high or simply zero dependent on whether a vehicle is assigned to the road segment or not. Using a time series of images acquired at the same time of day for several days, would average out this effect. However, long time series of aerial photos are costly to obtain. Hence, in order to remedy this effect, the use of graphs on the road network is able to average out the traffic estimates by considering the whole road network.

In Equation (1) T_i is an estimate for the number of vehicles using road segment i at a given time. A common metric for traffic is the AADT, which is the average number of daily passes, i.e. the average number of vehicles passing each day. In Larsen, Haug, and Aldrin (2008) a method of estimating the AADT based on very sparse traffic counts was presented. The approach extrapolates the instantaneous counts to a daily measure using an estimation of the daily traffic distribution which describes how the traffic changes during the day. For a given time, the instantaneous traffic count is converted to AADT by multiplying with a (time dependent) factor. In the following we describe a Traffic Denoising approach which also automatically finds this conversion factor.

4.2. *Traffic graph denoising*

4.2.1. Graphs, incidence matrices and graph Laplacian

A directed graph $\mathrm{G} = (\mathrm{N}, \mathrm{E})$ consists of a node set $\mathrm{N} = \{n_1, n_2, \ldots, n_N\}$ of cardinality N, and an edge set $\mathrm{E} = \{e_1, e_2, \ldots, e_E\}$ of cardinality E. For a directed graph we endow each edge e with an arbitrary reference orientation from its tail node $t(e)$ to its head node $h(e)$. The node-to-edge incidence matrix of a directed graph is defined as

$$B_{i,j} = \begin{cases} 1, & \text{if } h(e_j) = n_i, \\ -1, & \text{if } t(e_j) = n_i, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

The line graph of a graph G is a graph G_{LG} whose nodes correspond to the edges of G. Two nodes in G_{LG} are connected if the corresponding edges in the original graph G share an incident node. Given the incidence matrix $\mathbf{B}$ of the original graph, the adjacency matrix of the line graph can be expressed as $\mathbf{A}_{\mathrm{LG}} = |\mathbf{B}^{\mathrm{T}}\mathbf{B} - 2\mathbf{I}|$, where the absolute value is applied element-wise, T denotes the transpose, and $\mathbf{I}$ denotes the identity matrix Schaub and Segarra (2018). The graph Laplacian of the

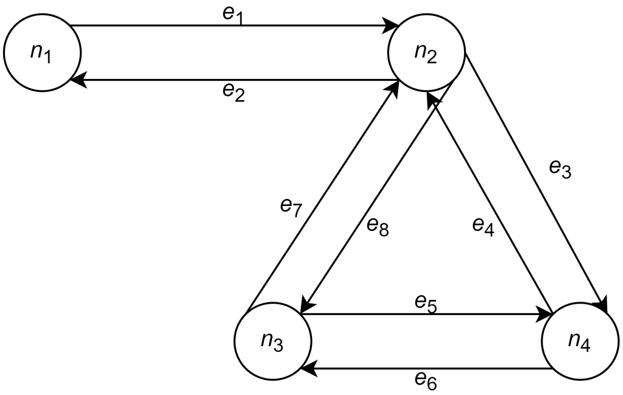


Figure 1.: Illustration of a two-way graph with edges modelling the flow in both directions between two given nodes.

line-graph is correspondingly defined as Schaub and Segarra (2018)

$$\mathbf{L}_{\mathrm{LG}} = \mathrm{diag}(\mathbf{A}_{\mathrm{LG}}\mathbf{1}) - \mathbf{A}_{\mathrm{LG}}, \tag{3}$$

where $\mathbf{1}$ denotes a vector of ones and $\mathrm{diag}(\cdot)$ denotes the diagonal matrix.

4.2.2. Laplacian filtering on graphs

Consider a noisy signal vector $\boldsymbol{y} = \boldsymbol{\mu} + \boldsymbol{\epsilon} \in \mathbb{R}^E$ defined on the edges of a connected graph G (or the nodes of the line graph G_{LG}), where $\boldsymbol{\mu}$ is the signal of interest and $\boldsymbol{\epsilon}$ represent zero-mean additive noise. We often assume that the signal $\boldsymbol{\mu}$ is smooth with respect to the underlying graph. Then, we can define the following filtering or denoising optimization problem

$$J = \arg\min_{\boldsymbol{\mu}} \left\{ \|\boldsymbol{y} - \boldsymbol{\mu}\|_2^2 + \alpha \boldsymbol{\mu}^{\mathrm{T}} \mathbf{L}_{\mathrm{LG}} \boldsymbol{\mu} \right\}, \tag{4}$$

where $\alpha > 0$ can be viewed as a regularization parameter that balances the influence of the smoothing due to the structure of the graph $\boldsymbol{\mu}^{\mathrm{T}} \mathbf{L}_{\mathrm{LG}} \boldsymbol{\mu}$ with the proximity to the noisy signal $\|\boldsymbol{y} - \boldsymbol{\mu}\|_2^2$. Please note that $\boldsymbol{\mu}^{\mathrm{T}} \mathbf{L}_{\mathrm{LG}} \boldsymbol{\mu}$ is a measure of the variation of the signal on the graph G_{LG} and is minimized by a constant vector proportional to $\mathbf{1}$. The optimal solution of the optimization problem is given by Schaub and Segarra (2018)

$$\widehat{\boldsymbol{\mu}} = (\mathbf{I} + \alpha \mathbf{L}_{\mathrm{LG}})^{-1} \boldsymbol{y} = \mathbf{F}\boldsymbol{y}, \tag{5}$$

where $\mathbf{F} = (\mathbf{I} + \alpha \mathbf{L}_{\mathrm{LG}})^{-1}$.

4.2.3. Traffic graphs and average annual daily traffic

We may use graphs to estimate the traffic flow along road segments in an area of interest. However, two-way traffic is often common. One way to model this is to use a directed graph, extended with an extra set of edges to model traffic in two directions (Figure 1). The input data now contains vehicle counts for each direction.

In a road network, we typically have traffic high-quality estimates of some road segments. These estimates are not obtained from EO data, but using other technologies like ground based counting stations. Let $\boldsymbol{d} = [d_1, d_2, \ldots, d_M]^{\mathrm{T}}$ denote the collection of these estimates. Then, we may constrain the graph estimates to these traffic estimates by including the constraint

$$\mathbf{C}^{\mathrm{T}} \boldsymbol{\mu} = \boldsymbol{d}, \tag{6}$$

where $\mathbf{C} = [\boldsymbol{e}_{c_1}, \boldsymbol{e}_{c_2}, \ldots, \boldsymbol{e}_{c_M}]$, $\boldsymbol{e}_k$ is an indicator vector equal to the kth column of the identity matrix, and the set $\{c_1, c_2, \ldots, c_M\}$ denotes the indices corresponding to the road segments where high-quality traffic estimates are available. We now define the cost function

$$J = \frac{1}{2}\left\|s\boldsymbol{y} - \boldsymbol{\mu}\right\|_2^2 + \frac{\alpha}{2}\boldsymbol{\mu}^{\mathrm{T}}\mathbf{L}_{\mathrm{LG}}\boldsymbol{\mu} + \left(\mathbf{C}^{\mathrm{T}}\boldsymbol{\mu} - \boldsymbol{d}\right)^{\mathrm{T}}\boldsymbol{\lambda}, \tag{7}$$

where $\boldsymbol{\lambda}$ is a vector of Lagrange multipliers, and then find the values of $\boldsymbol{\mu}$ and s that minimize

$$\widehat{s}, \widehat{\boldsymbol{\mu}} = \arg\min_{s,\boldsymbol{\mu}} J, \tag{8}$$

where s is a scale factor that adjusts the satellite derived AADT to the same level as the high-quality ones $\boldsymbol{d}$. The estimate of the scale factor is

$$\widehat{s} = \frac{\boldsymbol{y}^{\mathrm{T}}\boldsymbol{\mu}}{\boldsymbol{y}^{\mathrm{T}}\boldsymbol{y}}, \tag{9}$$

whereas to find the optimum value of $\boldsymbol{\mu}$ we compute the derivatives

$$\frac{\partial J}{\partial \boldsymbol{\mu}} = \mathbf{0} \quad \text{and} \quad \frac{\partial J}{\partial \boldsymbol{\lambda}} = \mathbf{0}. \tag{10}$$

This gives us an equation system that may be organized as

$$\begin{bmatrix} \mathbf{I} + \alpha\mathbf{L}_{\mathrm{LG}} & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} s\boldsymbol{y} \\ \boldsymbol{d} \end{bmatrix} \tag{11}$$

By solving the equation system Scharf (1991), we find that the solution for $\widehat{\boldsymbol{\mu}}$ is

$$\widehat{\boldsymbol{\mu}} = s\mathbf{F}\left(\mathbf{I} - \mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\mathbf{F}\mathbf{C}\right)^{-1}\mathbf{C}^{\mathrm{T}}\mathbf{F}\right)\boldsymbol{y} + \mathbf{F}\mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\mathbf{F}\mathbf{C}\right)^{-1}\boldsymbol{d}. \tag{12}$$

The matrix

$$\mathbf{P}_C = \mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\mathbf{F}\mathbf{C}\right)^{-1}\mathbf{C}^{\mathrm{T}}\mathbf{F} \tag{13}$$

is an oblique projection matrix onto the subspace $\langle\mathbf{C}\rangle$, i.e. the subspace spanned by the columns of $\mathbf{C}$. Let $\perp$ denote the orthogonal complement, i.e. $\langle\mathbf{C}\rangle^{\perp}$ is the orthogonal subspace of $\langle\mathbf{C}\rangle$, and $\mathbf{P}_C^{\perp} = \mathbf{I} - \mathbf{P}_C$ is an oblique projection matrix onto $\langle\mathbf{C}\rangle^{\perp}$. Then, since $\mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}$, we may express $\widehat{\boldsymbol{\mu}}$ as

$$\widehat{\boldsymbol{\mu}} = s\mathbf{F}\mathbf{P}_C^{\perp}\boldsymbol{y} + \mathbf{F}\mathbf{P}_C\boldsymbol{d}_{\mathrm{F}}, \tag{14}$$

where $\boldsymbol{d}_{\mathrm{F}} = \mathbf{F}^{-1}\mathbf{C}\boldsymbol{d} = (\mathbf{I} + \alpha\mathbf{L}_{\mathrm{LG}})\mathbf{C}\boldsymbol{d}$ is the high-quality traffic estimates, $\boldsymbol{d}$, filtered by the Laplacian matrix.

The challenge is that the estimates for s and $\boldsymbol{\mu}$ are not decoupled. We will therefore estimate s and $\boldsymbol{\mu}$ by alternating between estimating s using the previous estimate of $\boldsymbol{\mu}$ and estimating $\boldsymbol{\mu}$ using the previous estimate of s. We initialize this estimation procedure by setting $s = 0$ at the first iteration.

Please note that because of the scale factor s, the traffic estimates $\boldsymbol{\mu}$ will be calibrated towards the unit of the high-quality traffic estimates, $\boldsymbol{d}$. If these estimates are AADT estimates, the unit of $\boldsymbol{\mu}$ will also become AADT estimates.

The regularization parameter, α, is determined by cross-validation. For a given α, we subsequently remove one of the high quality AADT estimates, and estimate it from the graph denoising and aerial observations (Equation (14)). Then from all

	Number of identified vehicles	Number of detected vehicles	Precision	Recall
Number of driving vehicles	123	109	0.74	0.66
Number of parked vehicles	1 584	1 187	N/A	N/A

Table 1.: Precision and recall for driving and parked vehicles for the 2012 dataset.

high quality observations, we compute the root mean squared error (RMSE). The α with lowest RMSE is selected.

5. Results

5.1. *Vehicle detection*

To evaluate the performance of the Mask R-CNN based vehicle detection module, we computed the precision and recall for the driving vehicles for the city center part of the 2012 dataset (Table 1). In total 1707 vehicles were manually identified, and of them 1296 were detected. Among the 123 driving vehicles, 81 of them were correctly detected, 28 were parked vehicles falsely detected as driving, and 42 driving vehicles were falsely detected as parked. This resulted in precision and recall values of 0.74 and 0.66, respectively.

In Figure 2 and Figure 3 the vehicle detection and assignment are displayed. The detections from the neural network is described by the outline of the output segmentation colored in red in Figure 2 (a) and Figure 3 (a).

The further assignment of the detected vehicles to a road segment is shown in Figure 2 (c) and Figure 3 (c). Each vehicle is assigned to maximum one road segment: the nearest. As described in Section 3.3 this assignment is restricted both with regard to distance from road center line, vehicle to road angle and the FKB-veg database. In Figure 2 (c) and Figure 3 (c) the road segments are represented by the center line from the NVDB database. The road segments without any assigned vehicles are represented with a gray color, whereas the road segments with assigned vehicles are given an individual color which is also used to fill the assigned vehicle representations. Vehicles that are not assigned to any road segments are left unfilled and are only represented by their outline polygon.

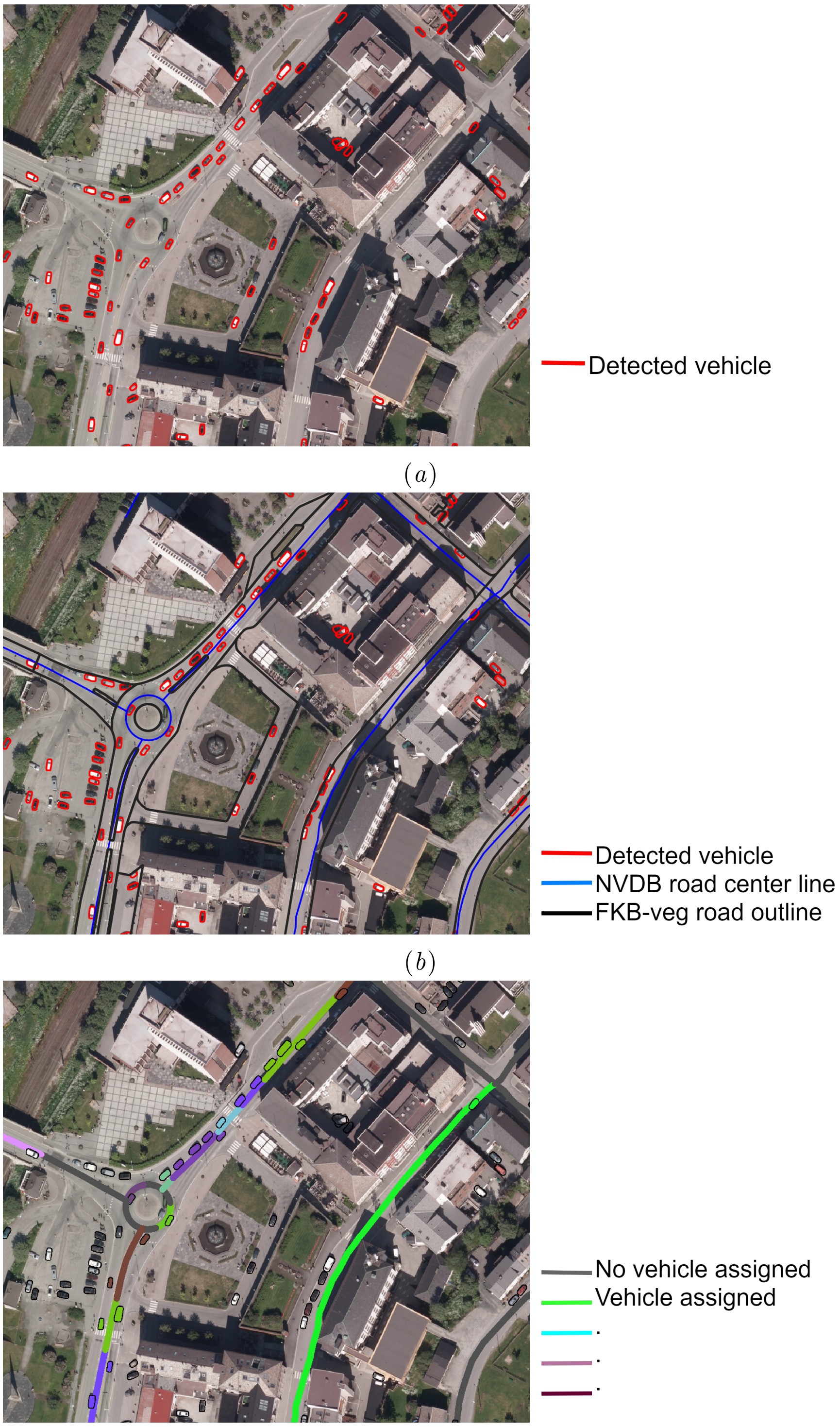
The detected vehicles were only assigned to a road segment if the distance from the center line was more than the estimated width of the road segment and the angle difference between the vehicle and the nearest portion of the road segment was less than 20°. The width of the road segments were estimated in a generous manner based on type of road (European, municipality, etc). In addition, the FKB-veg database were used to remove vehicles assumed to be parked along the side of the roads by shrinking the outline of the roads by 0.5 m and removing any vehicles intersecting the outline.

The vehicle detection was performed both on a dataset from Narvik 2012 and 2017, which enabled the calculation of an average count, thereby increasing the effective length of each road link by a factor of two. A heatmap of the resulting traffic estimation is shown in Figure 4 where road links with relative high traffic estimate is represented in dark red, whereas the low traffic road links are represented in a blue color.

5.2. *Annual average daily traffic estimation*

In Figure 4 we show an overview of the estimated traffic based on only the aerial images. The heatmap show that the traffic in the residential areas are quite low, whereas much of the traffic is located along the main road through the middle of the figure.

For the graph based processing the regularization parameter α was estimated to

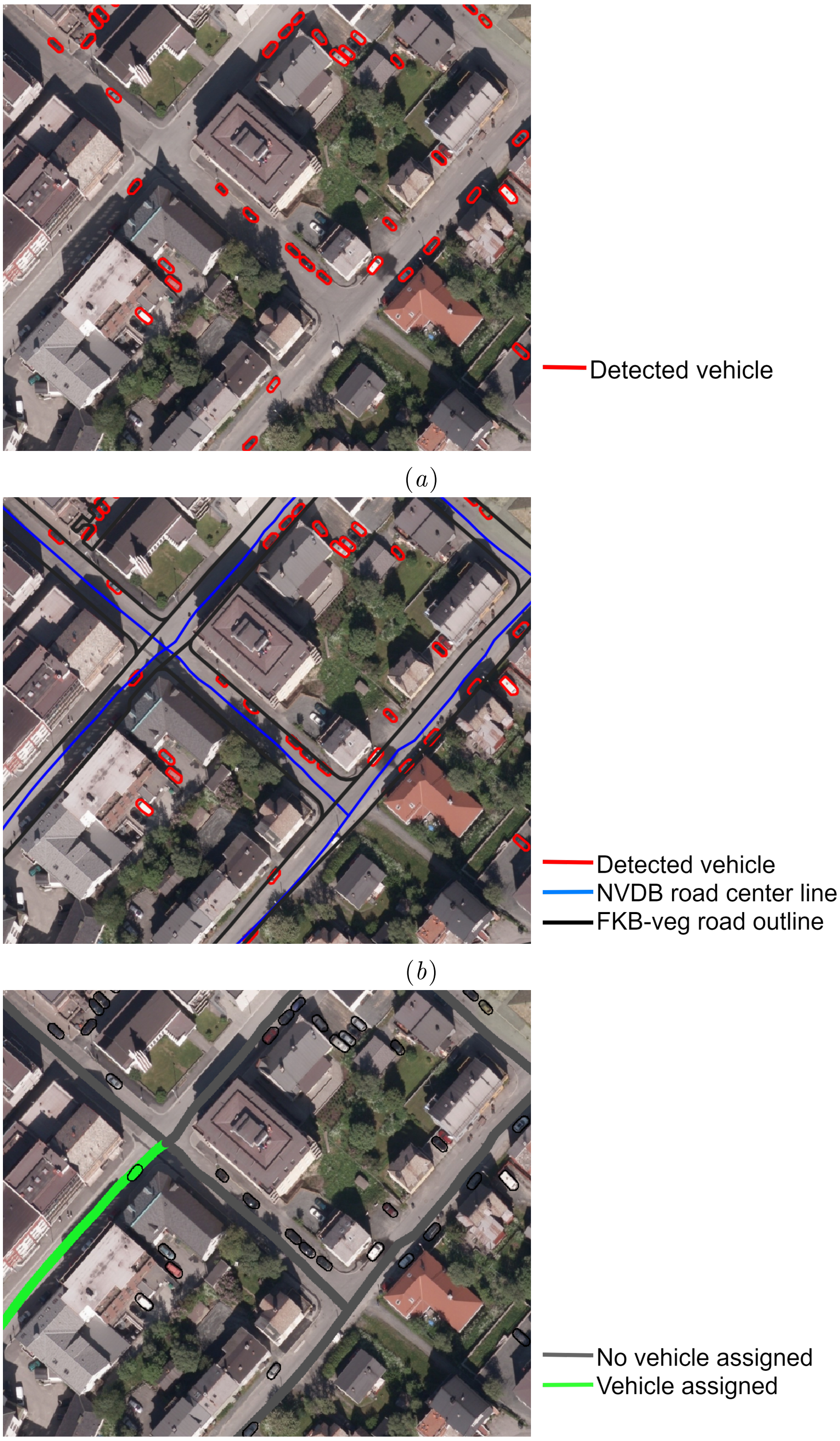


(a)

(b)

(c)

Figure 2.: Vehicle detections, city centre of Narvik, 2012. (a) Detected vehicles extracted from the output of the instance segmentation module (red). (b) Road links from the NVDB database represented by their center line (blue) and FKB-veg database road polygon outline (black). (c) Road segments without any assigned vehicles are represented with a gray color, and road segments with assigned vehicles are colored with an individual color which is also used to fill the assigned vehicles. Unfilled vehicles are not assigned to any road segment.



(a)

(b)

(c)

Figure 3.: Vehicle detections, zoom-in on some streets in Narvik, 2012. (a) Detected vehicles extracted from the output of the instance segmentation module (red). (b) Road links from the NVDB database represented by their center line (blue) and FKB-veg database road polygon outline (black). (c) Road segments without any assigned vehicles are represented with a gray color, and road segments with assigned vehicles are colored with an individual color which is also used to fill the assigned vehicles. Unfilled vehicles are not assigned to any road segment.

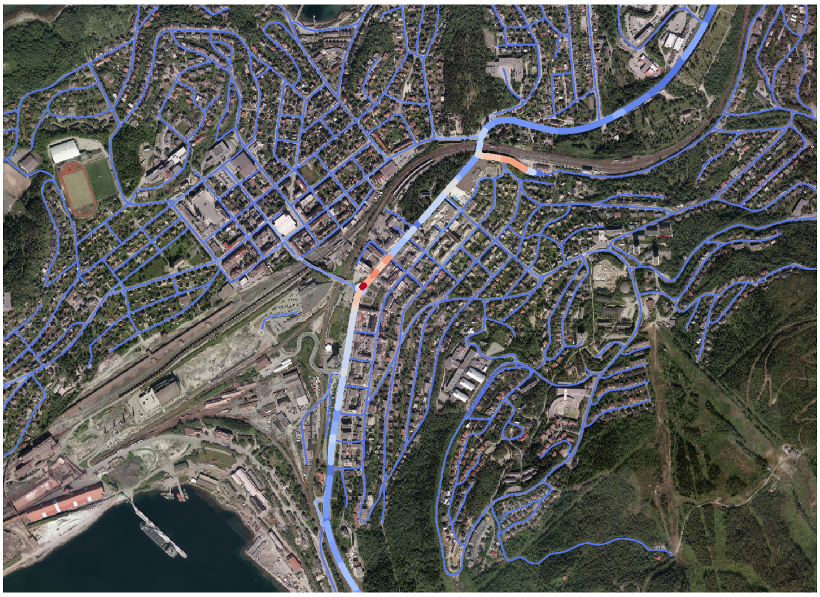


Figure 4.: Overview of traffic estimation in city centre of Narvik before graph smoothing. Road links with high amount of traffic are represented in red, whereas road links with low traffic are given a blue color.

	RMSE	Average bias
Proposed method	2 279	-383
Graph filtering applied to high quality traffic estimates	2 284	-411
No graph processing, only traffic estimates from aerial images	6 214	-4 170

Table 2.: RMSE and bias of AADT estimates.

200. By comparing the RMSE and bias of the proposed scheme with a scheme using only high-quality counts and graph processing, and a scheme using only traffic estimates from aerial images we observe that the proposed scheme slightly outperform the high-quality traffic estimates + graph processing scheme, and clearly outperform the traffic estimates based on aerial images only (Table 2). By performing a 20-fold cross validation on the high quality traffic counts we observe that the graph processing provides much smoother AADT estimates compared to the traffic estimates from aerial images only (Figure 5). The smoothing of AADT estimates are also reflected in the estimated heatmap (Figure 6).

6. Discussion

In Figure 2 (a) and Figure 3 (a) the vehicle detection is shown for a downtown area and a residential area in Narvik respectively. The vehicle detection performance of the neural network is good. For instance, the network is able to detect several vehicles located in shadows from buildings, a difficult task which often times even humans will struggle with. However, this was an area that was observed to benefit from the normalization with regard to road pixels (see. Section 3.1). In order to further improve the detection rate of vehicles in the shadow, it would be advantageous to have even more examples and variations in the training data. There are some vehicles that are not detected, however considering that the network is tested on a dataset not seen during training, the performance is satisfactory and the impact of these

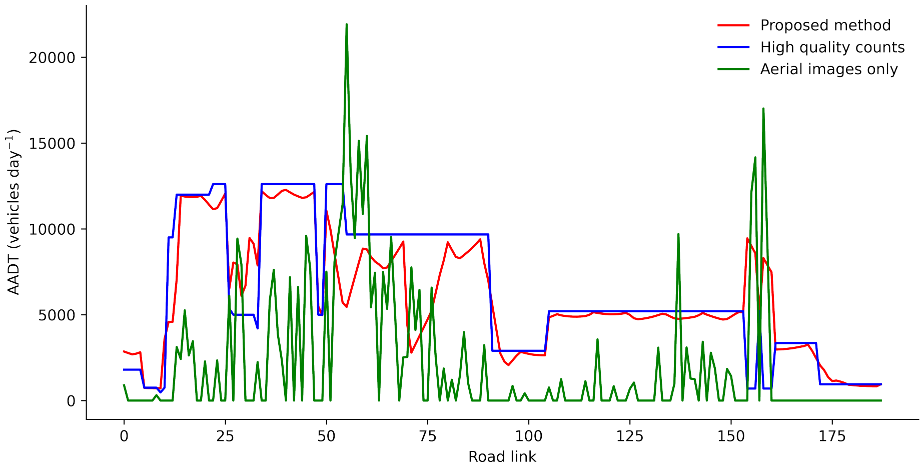


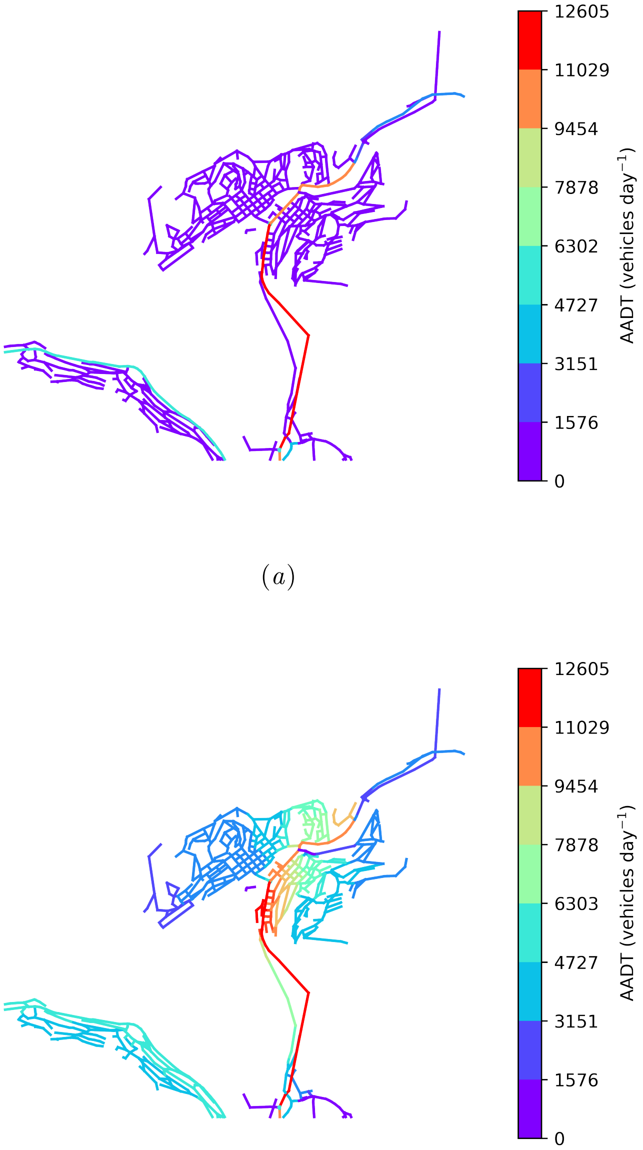
Figure 5.: AADT estimates using proposed method and 20-fold cross validation (red curve), AADT estimates based on only aerial images (green curve), and corresponding high-quality AADT estimates (blue curve).

false negatives are not the biggest concern. When it comes to false positives due to parked vehicles along the street, those have proved much more difficult to avoid, an error which may have a larger affect on the end result, especially in residential areas. Most of the vehicles detected in Figure 3 (a) seem to be parked along side the road, and thereby may be considered false detections.

In Figure 2 (b) and Figure 3 (b) both the FKB-veg road outlines (black) and the road link center lines (blue) from the NVDB database are depicted. It is apparent from these figures that the center line from the NVDB database are quite inaccurate, whereas the road edge is fairly well represented by the FKB-veg database. Note for instance in the lower part of Figure 3 (b) where the center line almost intersects the black outline polygon. Hence, using the FKB-veg outlines for removal of parked vehicles yields a significant improvement of the performance.

The mapping of detected vehicles to a road segment is shown in In Figure 2 (c) and Figure 3 (c). The road link segments are represented by their center line and segments with assigned vehicles are given a unique color, whereas segments without any assigned vehicles are shown in grey. Vehicles assigned to a road segment are filled with the same color as their respective road segment while unassigned vehicles are left unfilled. The mapping is fairly good; there are some falsely unassigned vehicles left of the roundabout in Figure 2 (c), however most of the vehicles seem to be correctly classified as moving or parked. Since the traffic estimation is to be used for distributing the total amount of traffic pollution, an important aspect is whether the parked vehicles in the residential areas are successfully filtered. The amount of parked vehicles in these areas are usually high while the amount of moving vehicles is nearly zero. Hence, an unsuccessful filtering of these vehicles, would erroneously shift the traffic distribution towards residential areas. In Figure 3 (c), the amount of parked vehicles are high, however, with the use of the FKB-veg procedure only one vehicle is assigned to a road link. By a visual inspection, this vehicle may very well in fact be a moving vehicle.

For many road links the number of counted vehicles was zero. The reason for this is that many of these road links are very short (only a few hundred metres), and we only have a time series of two aerial images covering the city. Hence, a given vehicle will only be visible in the road link for a short time period. There is also the problem of extremely short road segments receiving a very high estimated traffic measure based on a single assigned vehicle. Both of these issues would be improved by using a larger time series allowing an averaging of these noisy results. In effect, a larger time series would increase the length of each road segment and thereby increase the observation time of each road link, which in turn is expected to yield an improved quality of the aerial based traffic estimation.



(*a*)

(*b*)

Figure 6.: Road links with high amount of traffic are represented in red, whereas road links with low traffic are given in a purple color. (*a*) Overview of AADT per road link in Narvik for ground-based measurements. (*b*) Overview of AADT per road link in Narvik obtained using the proposed graph smoothing method.

Graph smoothing was necessary in order to provide consistent AADT estimates for the road network, in particular with only two aerial observations of each road link. Without graph smoothing, the RMSE was much higher (Table 2), and the AADT estimates varies substantially between two consecutive road links (Figure 5). The noisiness of the aerial based AADT estimates resulted in a high value of the regularization parameter α, which again resulted in very smooth graph estimates.

For our case, we observed that the impact of the aerial based traffic estimates was minor (Table 2). This is related to the noisy observations from the aerial images, which resulted in a very large regularization parameter α. By constraining the graph output estimations on the high quality road link AADT estimates, the strong regularization of the graph resulted in that these AADT estimates had a huge impact to the graph estimates on all the other road links.

In Larsen et al. Larsen, Haug, and Aldrin (2008), the authors investigated whether traffic observations of only a few minutes were sufficient to obtain acceptable estimates of the AADT. By simulating traffic counts at 5 min resolution from a set of ground-based traffic stations, they showed even with limited data, valuable information may be provided despite the uncertainty involved. For an urban environment, with a road link length of 300 m and a speed limit of 30 km h^{-1}, at least 5 aerial images are needed to provide a 5 min measurement interval.

7. Conclusions

With the introduction of deep learning into remote sensing, we have demonstrated that estimating AADT from VHR optical remote sensing images is possible. However, in order for this to be achievable there are some requirements: Firstly, a high quality road mask (like our FKB-veg road mask) is necessary in order to separate driving vehicles from parked ones. Secondly, depending on the length of the road links, several remote sensing images are necessary in order to increase the observation time of each road link, and finally, graph smoothing is necessary in order to obtain consistent and less noisy estimates across the road network. The advantage of the proposed methodology is automatic AADT estimation from VHR optical remote sensing images, which at the same time utilizes high-quality ground-based AADT estimates for road links where such estimates exist.

For future work we will focus on further improve the separation of driving vehicles from parked ones, and improving the graph processing by considering other data distributions that are able to model zero-inflated observations.

Acknowledgments

Thanks to Michael Gauss and Bruce Denby at the The Norwegian Meteorological Institute for fruitful and interesting discussions during the AirQuip project period.

Disclosure statement

The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Funding

This research was funded by the Research Council of Norway, grant number 267734 (AirQuip project).

References

Audebert, N., B. Le Saux, and S. Lefèvre. 2017. "Segment-before-detect: Vehicle detection and classification through semantic segmentation of aerial images." *Remote Sensing* 9 (368).

Eikvil, L., L. Aurdal, and H. Koren. 2009. "Classification-based vehicle detection in high-resolution satellite images." *ISPRS J. Photogramm. Remote Sensing* 64 (1): 65–72.

He, Kaiming, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. 2018. "Mask R-CNN." *arXiv preprint arXiv:1703.06870v3* .

Hinz, S., and U. Stilla. 2005. "Detection of vehicles and vehicle queues for road monitoring using high resolution aerial images." In *9th World Multiconference on Systemics, Cybernetics, and Informatics*, Orlando, Fl.

Int. Soc. Photogramm. Remote Sensing (ISPRS). 2018. "2D Semantic Labeling Contest." online. http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html.

Kampffmeyer, M., A. B. Salberg, and R. Jenssen. 2016. "Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks." In *Proc. IEEE Conf. Computer Vision Pattern Recognition Workshops*, 1–9.

Krizhevsky, A., I. Sutskever, and G. E. Hinton. 2012. "ImageNet classification with deep convolutional neural networks." In *NIPS*, 1106–1114.

Larsen, S. Ø., O. Haug, and M. T. Aldrin. 2008. *Estimating Annual Average Daily Traffic (AADT) based on extremely sparse traffic counts - a study of the feasibility of using satellite data for AADT estimation.* NR-Note SAMBA/08/49. Oslo: Norwegian Computing Center.

Larsen, S. Ø., A. B. Salberg, and L. Eikvil. 2013. "Automatic system for operational traffic monitoring using very-highresolution satellite imagery." *Int. J. Remote Sensing* 34 (13): 4850–4870.

Lato, M., R. Frauenfelder, and Y. Bühler. 2012. "Automated detection of snow avalanche deposits: Segmentation and classification of optical remote sensing imagery." *Natural Hazards Earth Syst. Sci.* 12 (9): 1–14.

Leitloff, J., S. Hinz, and U. Stilla. 2010. "Vehicle detection in very high resolution satellite images of city areas." *IEEE Trans. Geosci. Remote Sens.* 48 (7): 2795–2806.

Leitloff, J., D. Rosenbaum, F. Kurz, O. Meynberg, and P. Reinartz. 2014. "An operational system for estimating road traffic information from aerial images." *Remote Sens.* 11: 11315–11341.

Liu, K., and G. Mattyus. 2015. "Fast multiclass vehicle detection on aerial images." *IEEE Geosci. Remote Sens. Lett.* 12 (9): 1938–1942.

Long, J., E. Shelhamer, and T. Darrell. 2015. "Fully convolutional networks for semantic segmentation." In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, 3431–3440.

Lv, Z. Y., T. F. Liu, P. Zhang, J. A. Benediktsson, T. Lei, and X. Zhang. 2019. "Novel Adaptive Histogram Trend Similarity Approach for Land Cover Change Detection by Using Bitemporal Very-High-Resolution Remote Sensing Images." *IEEE Trans Geosci. Remote Sensing* 57 (12): 9554–9574.

Massa, F., and R. Girshick. 2018. "maskrcnn-benchmark: Fast, modular reference implementation of Instance Segmentation and Object Detection algorithms in PyTorch." https://github.com/facebookresearch/maskrcnn-benchmark. Accessed: 2019.

Paisitkriangkrai, S., J. Sherrah, P. Janney, and A. Hengel. 2015. "Effective semantic pixel labelling with convolutional networks and conditional random fields." In *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops*, 36–43.

Sabour, S., N. Frosst, and G. E. Hinton. 2017. "Dynamic routing between capsules." In *Adv. Neural Inform. Process. Syst.*, 1–11. Long Beach, Calif.

Scharf, L. L. 1991. *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis.* Reading, Mass.: Addison-Wesley.

Schaub, M. T., and S. Segarra. 2018. "Flow smoothing and denoising: graph signal processing in the edge-space." *arXiv preprint arXiv: 1808.0211* .

Wind, P., and B. R. Denby. 2017. *Transboundary particulate matter, photo-oxidants, acidifying* *and eutrophying components. EMEP Status Report 1/2017*, Chap. Local Fractions in the EMEP MSC-W model, 77–86. Oslo, Norway: The Norwegian Meteorological Institute.

Yu, Y., H. Ai, X. He, S. Yu, X. Zhong, and M. Lu. 2018. "Ship Detection in Optical Satellite Images Using Haar-like Features and Periphery-Cropped Neural Networks." *IEEE Access* 6 (doi: 10.1109/ACCESS.2018.2881479.): 71122–71131.

Yu, Y., T. Gu, H. Guan, D. Li, and S. Jin. 2019. "Vehicle detection from high-resolution remote sensing imagery using convolutional capsule networks." *IEEE Geosci. Remote Sens. Lett.* 16 (12): 1894–1898.

Zhao, T., and R. Nevatia. 2003. "Car detection in low resolution aerial images." *Image and Vision Computing* 21: 693–703.

Zhao, Z., P. Zheng, S. Xu, and X. Wu. 2019. "Object detection with deep learning: a review." *IEEE Trans. Neural Networks Learning Syst.* 30 (11): 3212–3232.

List of Figures

List of Tables