Urban Water Journal

ISSN: 1573-062X (Print) 1744-9006 (Online) Journal homepage: https://www.tandfonline.com/loi/nurw20

Automated localization of urban drainage infrastructure from public-access street-level images

Dominik Boller, Matthew Moy de Vitry, Jan D. Wegner & João P. Leitão

To cite this article: Dominik Boller, Matthew Moy de Vitry, Jan D. Wegner & João P. Leitão (2019) Automated localization of urban drainage infrastructure from public-access street-level images, Urban Water Journal, 16:7, 480-493, DOI: 10.1080/1573062X.2019.1687743

To link to this article: https://doi.org/10.1080/1573062X.2019.1687743

Published online: 11 Nov 2019.

URBAN WATER JOURNAL
2019, VOL. 16, NO. 7, 480–493
https://doi.org/10.1080/1573062X.2019.1687743

Taylor & Francis
Taylor & Francis Group

RESEARCH ARTICLE

Automated localization of urban drainage infrastructure from public-access street-level images

Dominik Boller[a], Matthew Moy de Vitry[a,b], Jan D. Wegner[b] and João P. Leitão[a]

[a]Department of Urban Water Management, Eawag: Swiss Federal Institute of Aquatic Science and Technology, Dübendorf, Switzerland; [b]EcoVision Lab, ETH Zurich, Zurich, Switzerland

ABSTRACT

Comprehensive management of urban drainage network infrastructure is essential for sustaining the operation of these systems despite stresses from component deterioration, urban densification, and a predicted intensification of rainfall events. In this context, up-to-date and accurate urban drainage network data is key. However, such data is often absent, outdated, or incomplete. In this study, a new approach to localize manhole covers and storm drains, using deep learning to mine publicly available street-level images, is presented, tested, and assessed. Thus, the time-consuming and costly acquisition of the location of these system components can be avoided. The approach is evaluated using 5,000 high-resolution panoramas covering 500 km of public roads in Switzerland. The object detection approach proposed shows good performance and an improvement over state of the art image-based urban drainage infrastructure component detection. While the geographical localization of the detected objects still contains errors, the accuracy achieved is nevertheless sufficient for some applications, e.g. flood risk assessment.

ARTICLE HISTORY
Received 10 January 2019
Accepted 16 October 2019

KEYWORDS
Deep learning; highresolution streetlevel imagery; urban drainage system components

1. Introduction

The sustainable development and management of urban drainage systems depends on knowledge of the number, location and characteristics of the components forming such systems. However, even in countries with highly developed infrastructure such as Switzerland, complete and accurate urban drainage system infrastructure data are often not present or not easily available (Maurer et al. 2012). This lack of information, e.g. not knowing where drain inlets are located nor their hydraulic characteristics, hinders comprehensive infrastructure asset management. The lack of information is alarming, when considering the immense replacement cost of urban drainage infrastructure (in Switzerland it amounts to 100 billion Swiss Francs (Hoffmann, Hunkeler, and Maurer 2014), which corresponds to approximately 10,700 Euros per inhabitant). Additionally, most urban construction work requires detailed and accurate maps of underground infrastructure to prevent damage and service interruptions. For example, Agadakos et al. (2013) identified the need for real-time access to urban underground infrastructure data and developed a comprehensive decision making tool aiming at supporting local construction interventions in the public space.

Besides facilitating infrastructure management, detailed information on drainage system network topology is also essential for simulating the hydraulic behavior of drainage systems. Simulation results can be used to improve the operation of wastewater treatment plants (Seggelke et al. 2005) or to assess flood risk and evaluate mitigation options. For example, Hürter and Schmitt (2016) found that the consideration of drain inlets in pluvial flood models is especially important for moderate rain events. Moreover, it has been shown that the location and cleanliness of drain inlets significantly influences drainage and flood behavior in urban areas (Chang et al. 2018; Leitão et al. 2017). With a predicted increase in flood risks in many regions due to climate change (Hirabayashi et al. 2013) and increased urbanization (Nirupama and Simonovic 2006), accurate urban flood modelling tools can be relevant to help understanding the future flood scenarios.

In the complete absence of urban drainage system network information, the surface elements of these systems (namely manhole covers and drain inlets), can serve as reference points for inferring the location and topology of the network. This was demonstrated by Commandré et al. (2017a), who used the location of manhole covers in combination with digital elevation models to estimate the topology of urban drainage system networks. Conventionally, drain inlets and manhole covers are mapped during manual field surveys (hydraulic attributes are usually also assigned to the components during these surveys), which are both time-consuming and expensive.

In this study, a novel and scalable approach to automatically identify and locate drain inlets and manhole covers, solely based on publicly available images, is proposed. The approach is possible thanks to three recent developments: (i) the emergence of national and global street-level imagery services, (ii) the improvement of convolutional neural networks (CNNs) architectures and their training processes for object classification and detection, and (iii) the general availability of high-power computing. To test the feasibility and viability of this approach on a regional scale, Google Street View data covering 500 km of public roads from five municipalities in Switzerland are analyzed.

2. Previous related work

In the past 20 years, the amount of data collected in urban environments has proliferated, spurring the development of

CONTACT João P. Leitão joaopaulo.leitao@eawag.ch

novel approaches for the localization of urban infrastructure components. In addition, developments in computer processor technology have led to a maturation of machine learning and especially deep learning as modelling tools (Sejnowski 2018). According to Voulodimos et al. (2018), deep learning methods have been shown to outperform previous state-of-the-art machine learning techniques in several fields, namely on computer vision tasks. In previous research, hand-crafted filters and machine-learning approaches with varying degrees complexity have been applied to the problem of urban drainage infrastructure mapping. In each case, the specific method applied often depends on the type and quality of data available.

The most common methods found in the literature are based on georeferenced aerial images. Niigaki, Shimamura, and Morimoto (2012) detected manhole covers based on separability and uniformity of feature distributions using the Bhattacharyya coefficient (i.e. an approximate measurement of the amount of overlap between two statistical samples). Also, by means of a circular geometric filter, Bartoli et al. (2015) detected manhole covers in very high resolution aerial images. This approach was further developed by Pasquet et al. (2016) who combined it with a machine learning technique to build a model to detect manhole covers. More recently, Commandré et al. (2017b) applied a customized CNN to the task, achieving the best results published to date for manhole covers. For detecting drain inlets, Moy de Vitry et al. (2018) proposed a multiview approach based on images taken from an unmanned aerial vehicle (UAV). The authors used a Viola Jones type classifier (Viola and Jones 2001) and moving window for detection, followed by clustering and cluster classification steps to consolidate information from multiple views. Nevertheless, the use of aerial images for manhole localization has two main disadvantages that are intrinsic to the approach: (i) the typical resolution of the highest-resolution images (around 5–10 cm/pixel) is still very low when compared to the typical size of manholes and drain inlets (50–100 cm), and (ii) there is a large likelihood of the objects being hidden by trees and vehicles. While hyperspectral airborne images provide useful information for mapping urban road infrastructure (Herold et al. 2003), the authors are not aware of such data being used for detecting objects in urban areas.

Another approach is based on data collected at street level with Light Detection And Ranging (LiDAR) technology. For instance, Yu et al. (2015, 2014) have already achieved very good results detecting manhole covers and drain inlets with LiDAR data, which was collected from the ground in a very high resolution. However, LiDAR data collection is still expensive and the processing of LiDAR data is complicated when compared to image processing, which puts this approach at a disadvantage.

The use of street-level images is also a possible way to detect and locate drainage system manhole covers and stormwater drain inlets. These images are are often collected with a car-mounted camera rig. Timofte and van Gool (2011) were the first to pursue this approach for mapping manholes in images recorded from a moving van equipped with several cameras. With this set-up, manholes were observed in multiple images and from multiple viewpoints, thereby increasing the likelihood of detection. At the same time, they faced many challenges such as regular changes in illumination conditions and substantial viewpoint variance given different positions on the road. While Timofte and van Gool (2011) performed manhole cover detection with images collected specifically for that purpose, it is also possible to use images collected by mapping services such as Google Street View, Baidu, Yandex, Mapillary and OpenStreetCam. In doing so, data collection can be performed wherever the service is provided and costs can be significantly reduced. Demonstrating the potential of this approach, Hebbalaguppe et al. (2017) presented a method for updating assets for telecommunication infrastructure using Google Street View images of 2,048 × 2,048 px resolution and a deep learning model. Recently, Krylov, Kenny, and Dahyot (2017) also used Google Street View images to automatically detect and geo-locate traffic lights and telegraph poles, achieving high object recall rates and localization accuracy within 2 meters. While these studies share certain similarities with the present work, the problem addressed in this study is of particular difficulty due to the small size and typical locations of drain inlets and manholes. The preferred use of Google Street View images for these studies in comparison to images from other street-level image repositories is explained by the less variability of Google Street View images quality and their higher resolution. Nevertheless, the approach presented in this study would work with images from other street-level image repositories.

3. Automatic object detection and localization framework

3.1. Overview

The methodology presented in this study (Figure 1) is based on the work of Wegner et al. (2016) and Branson et al. (2018), which aimed to detecting urban trees and recognizing their species from street-level panoramas combined with aerial imagery. The good results obtained in those two studies motivated us to consider the methodology and adjust it to the specific case of the identification and localization of urban drainage systems components, such as manhole covers and stormwater drain inlets. First, street-level panoramas within a region of interest are downloaded, along with their metadata. Then, manhole covers and drain inlets are detected within the individual panoramas. Finally, with the panorama metadata, the geographical locations of the detected objects are estimated.

3.2. Data collection

Time-consuming and costly field surveys can be avoided by utilizing data from street-level imagery services. In the case of Google Street View, high-resolution, 360° panoramas can be downloaded via the Google Street View Static API[1]. In this study, a Python script was written to identify and download all panoramas and their metadata within a user-defined region via the Google Street View Static API. In a first approximation, it can be assumed that the projection used to create the panoramas is cylindrical. The metadata provided with each panorama includes the information about the vehicle position, as well as the yaw and tilt of the camera rig at the moment the images for the panorama were taken.

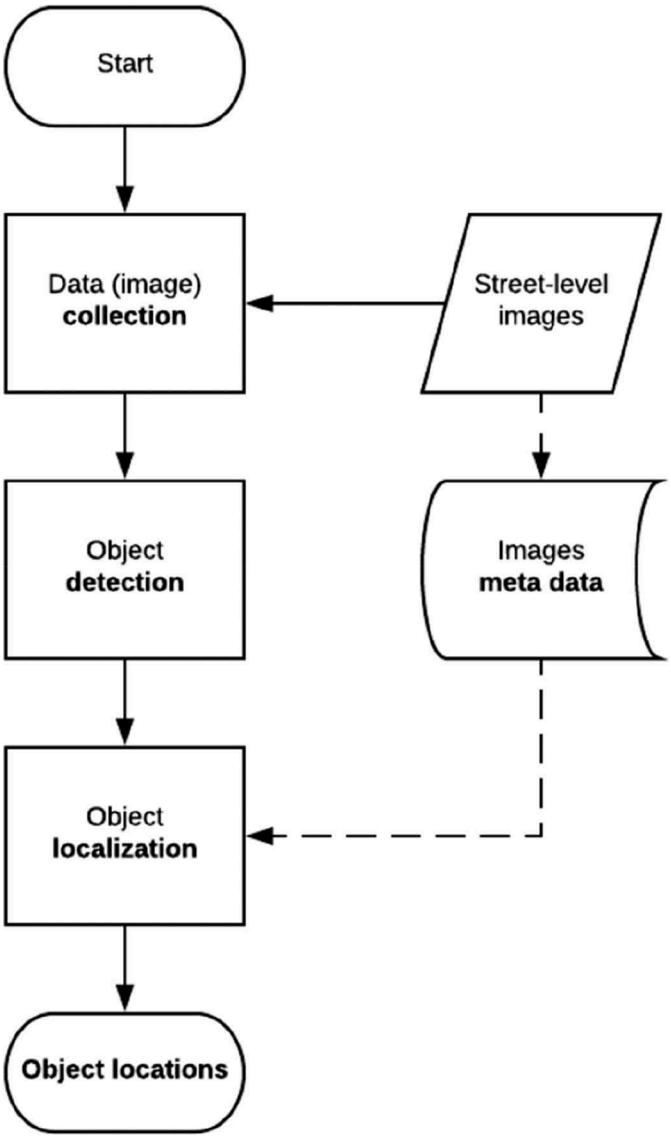


Figure 1. Automatic framework to detect stormwater drain inlets and manhole covers in street-level imagery and estimate their geographic location.

3.3. The proposed approach to detect urban drainage elements from street-level images

Object detection is the task of locating and classifying all instances of known object classes within an image. Automatic object detection in images starts with the extraction of object-specific visual features, which can be used to predict an object's presence and location within the image. Prior to deep learning, researchers manually engineered these image features according to their knowledge and observations of the object's appearance. Today, according to Guo et al. (2016) CNNs are the most suitable deep learning techniques for images, and given sufficient training data, CNNs are able to learn these features directly from annotated images.

3.3.1. Object detection model architecture

CNNs consist of multiple layers that consecutively extract features, starting with the raw pixel values of the image. As the image data advances through the network's layers, the extracted features become increasingly specific and discriminative. Based on the last feature layer, the final image class is predicted. The task of object detection is commonly distinguished in two parts: the localization of the object within the image (region proposal), and the subsequent classification of the proposed region. Region suggestion can be done in different ways, ranging from simple methods such as a sliding window to advanced methods such as using a region proposal network (RPN). The RPN was introduced with Faster R-CNN (Ren et al. 2017) and currently represents a state of the art methodology for region proposals. It shares convolutional layers with the CNN used for classification and therefore reduces costs for computing proposals significantly. Based on the shared convolutional features RPN regresses region bounds and objectness scores thereby identifying region proposals. The RPN and the CNN for classification can both be trained end-to-end with supervised learning (i.e. learning a function that maps an input to an output using example input-output pairs). During training, region proposal and classification are optimized in a parallel and synergistic manner, thereby unifying the network's internal parameters.

Presently, a great number of modern convolutional object detectors variants have been proposed, each developed for a specific application. For example, the architectures used for real-time detections tend to have a simpler, more shallow architecture (e.g. less layers) to generate faster predictions. Deeper models, on the other hand, tend to achieve higher accuracy (given sufficient training data), since they can extract more complex and class-specific features from the images. This increase in accuracy comes at the expense of speed and vice versa. Huang et al. (2017) evaluated this speed/accuracy trade-off for a selection of the current state of the art object detection models. Since speed is not critical for the task at hand in this study, as it is not a real-time application meaning that we can afford a few days to obtain the results, an object detection model with one of the highest detection accuracies was selected: the Faster R-CNN as meta-architecture and Resnet 101 (He et al. 2016) as feature extractor. The model's output consists of a bounding box specifying the object's boundaries, a label specifying the object's class and confidence score indicating the model's confidence in its prediction (details on the calculation of the confidence score are, for example, presented in the *Convolutional Neural Networks for Visual Recognition* course notes, available at http://cs231n.github.io/linear-classify/#softmax).

3.3.2. Tiling and stitching of images and annotations

By default, the selected model architecture rescales images to a fixed size in order to keep the computational and memory load under control. In the case of very high resolution panoramas, this rescaling would lead to a loss of information that is necessary for detection of small objects, such as manhole covers and drain inlets that have diameters ranging from 50 to 100 cm. These objects are relatively small in the field of view of a full panorama and require a high level of image detail to be detected (Huang et al. 2017). Modifying the CNN to the resolution of the panorama would exhaust the computing resources. Therefore, in this study the panoramas were tiled into smaller

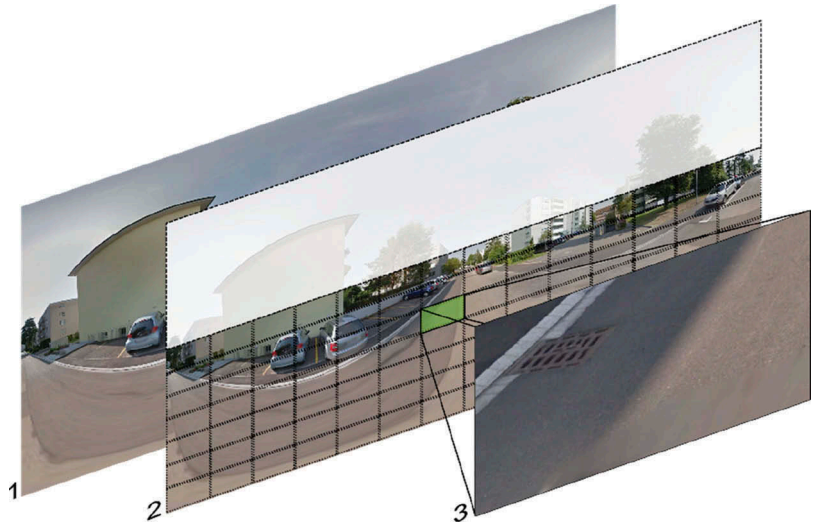


Figure 2. The original panorama (1) was cropped to remove the sky, while its remaining part was split into tiles with horizontal and vertical overlap (2). The resulting tiles were fed into the object detection model individually (3).

images, and each tile was fed into the object detection model separately (Figure 2).

Naturally, the annotations used for training and testing the model must also be clipped. There is a negative side-effect of this solution, since some objects and their annotations will be clipped as well, sometimes leading to low-quality annotations (e.g. annotations for barely visible objects at the edge of the tile) that could mislead the training and evaluation of the detection model. To counter this issue, the following criteria (Equations 1, 2 and 3) were introduced to remove such annotations.

$$width\ of\ tiled\ bounding\ box < \beta_{width}, \tag{1}$$

$$height\ of\ tiled\ bounding\ box < \beta_{height}, \tag{2}$$

$$\frac{area(tiled\ bounding\ box)}{area(original\ bound\ box)} < \beta_{area}, \tag{3}$$

where β_{width} is the minimal width of the tiled bounding box, β_{height} is the minimal height of the tiled bounding box, and β_{area} is the minimal area ratio of the tiled bounding box. The parameters in Equations 1–3 were not tuned as: (i) it is not within the objective of the present study; (ii) would require retraining (a long process) and evaluating the model several times, and (iii) the (expected) outcome would be a small performance increase. The set of values for these parameters are presented in Table B1 (in Appendix B).

Based on the camera's orientation, assumed to be parallel to the ground, the top part of the images would be sky or horizon, which are not of interest for identifying manholes or drain inlets. By looking at different images, the top 600 px of the images were cropped, so that the image area to be analysed is reduced.

3.4. Localization

To project coordinates from image space to geographic coordinates, a method specific to street-level 360° panoramas, proposed by Wegner et al. (2016), was used.

3.4.1. Projection of image coordinates into real world coordinates

The previously detected manhole covers and drain inlets are represented by the image coordinates (x, y) of the bounding box centers. Under the assumption of known camera height and locally flat terrain, one can estimate the relative position of an object's location with respect to the position of the camera (lat_c, lng_c) based on Equations 4, 5 and 6.

$$e_x = sinsin\left(x\frac{2\pi}{W} - \pi + yaw_c\right)z \tag{4}$$

$$e_y = coscos\left(x\frac{2\pi}{W} - \pi + yaw_c\right)z \tag{5}$$

$$z = \frac{-h}{tan\left(-y\frac{\pi}{H} + \frac{\pi}{2}\right)} \tag{6}$$

where e_x and e_y are the metric distances of the object from the camera in the east and north directions, z is the planar distance from the object to the camera, h is the height of the camera above the ground, $W \times H$ are dimensions of the panorama in pixels, and yaw_c is the camera's heading. To obtain the position of the object in real-world coordinates (lat, lng), it is sufficient to transform the metric distances into relative latitude and longitude using the earth radius R and add them to the location of the camera (Equations 7 and 8).

$$lat = lat_c + arcsin\left(e_y, R\right) \tag{7}$$

$$lng = lng_c + arcsin\left(\frac{e_x}{cos(lat_c)}, R\right) \quad (8)$$

3.4.2. Non-maximum suppression

The overlap between street-level images, as well as panorama stitching artefacts and the buffered panorama tiling can cause the same objects to be detected and then localized multiple times. Non-Maximum Suppression (NMS) is the process of removing any non-dominant detections, retaining a single detection and location for each object. In this study, a greedy NMS algorithm is implemented: given a set of predicted object locations, the process starts by identifying the detection with the highest confidence score and marking it for retention. All surrounding detections within the specified search radius are then suppressed. Of the remaining detections, the one with the highest confidence score is identified and the process is repeated until all detections have either been suppressed or marked for retention. Although this approach is somehow computational demanding, it is significantly smaller when compared to the computational demand of training the faster R-CNN.

4. Experiment

4.1. Data

4.1.1. Street-level panoramas and covered regions

Google Street View is one of the most widespread street-level imagery services available, providing a homogeneous and dense coverage of urban and suburban areas in many countries around the world. In the USA and the UK for example, 99% of public roads have been covered by Google Street View. In Switzerland, Google Street View panoramas are captured approximately every 10–15 m for urban and suburban roads, and at larger intervals for motorways. The panoramas are created by capturing images with a multi-camera rig, often mounted on the roof of a car at a height of around 3 m. The images are then stitched together into 360° panoramas of apparently cylindrical projection.

To test the robustness of our framework at a regional level, 5,000 Google Street View panoramas of very high resolution (13,312 × 6,656 px) were downloaded for five municipalities in the canton of Zurich, Switzerland (Figure 3). Each panorama file is between approximately 3 and 8 MB in file size. For each municipality, 1,000 panoramas were downloaded (see Figure A1 in Appendix A for example panoramas), covering in total approximately 500 km of roads. The selected regions contain various types of drain inlets and manholes, which is assumed to be representative of the diversity found at a regional scale. The panoramas also contain variety in terms of lighting and season, since the images were captured during different periods of time.

4.1.2. Manual panorama annotations

The panoramas downloaded in the previous step were manually annotated by marking objects of interest with a bounding box and labelling each bounding box with its corresponding class (manholes, drain inlets, and water supply network valves). Valves were annotated as well so that the object detection model can better learn the differences between the manholes and valves, resulting in better performance for manholes.

Annotations were only made for objects that could be identified from their appearance, even if a distant object could sometimes be correctly identified by a human due to their placement on the road or sidewalk. Thus, the complexity of the detection problem is limited to focusing on object appearance and not context. Thanks to the spatial frequency of the panoramas, far-off objects in the images can be disregarded since often, another panorama is situated closer to the object for detection. By setting these bounds to the detection problem, the rate of human errors made while annotating the data is also reduced. The 5,000 panoramas used in this study were annotated by an external company, corresponding to a total of 8,970 manhole covers, 6,714 drain inlets and 4,456 valves. The annotation took approximately 167 hours.

4.2. Division of data into intra- and extra-regional datasets

To assess the real world applicability and scalability of the presented approach, it was investigated to what extent the object detection model, trained on images from one region, could be applied to images from that region or from nearby regions. The panoramas were divided into two geographically distinct groups, one for training, validation, and testing (intra-regional datasets) and the other just for testing (extra-regional dataset). The number of images used for training is the largest subset of the total images in order to increase the performance of the model; according to Sun et al. (2017) the performance on vision tasks increases logarithmically based on volume of training data size.

4.2.1. Intra-regional datasets

The intra-regional datasets (for training, validation, and testing) are formed by the annotated 4,000 panoramas for the city of Zurich and the municipalities of Dübendorf, Fehraltorf and Uster. The panoramas are randomly distributed into training, validation and test datasets. The training dataset, with 3,060 panoramas (i.e. 76.5%), is used to learn the internal parameters of the object detection model, while the validation dataset of 540 panoramas (i.e. 13.5%) is used to monitor and tune the training process. The test dataset contains the remaining 400 panoramas (i.e. 10%) and was used to evaluate object detection performance after training was completed.

4.2.2. Extra-regional dataset

The extra-regional dataset contains the 1,000 annotated panoramas for Adliswil, another municipality in the Canton of Zurich (Figure A1). All panoramas in this dataset were used to test the performance of the object detection model trained with the intra-regional training dataset.

4.3. Training the object detection model

Model training is an iterative search for model parameters that minimize the deviation (total loss as defined, for example, in Barber (2012)) of the model detections from the ground truth, represented in this study by the annotations. This search was

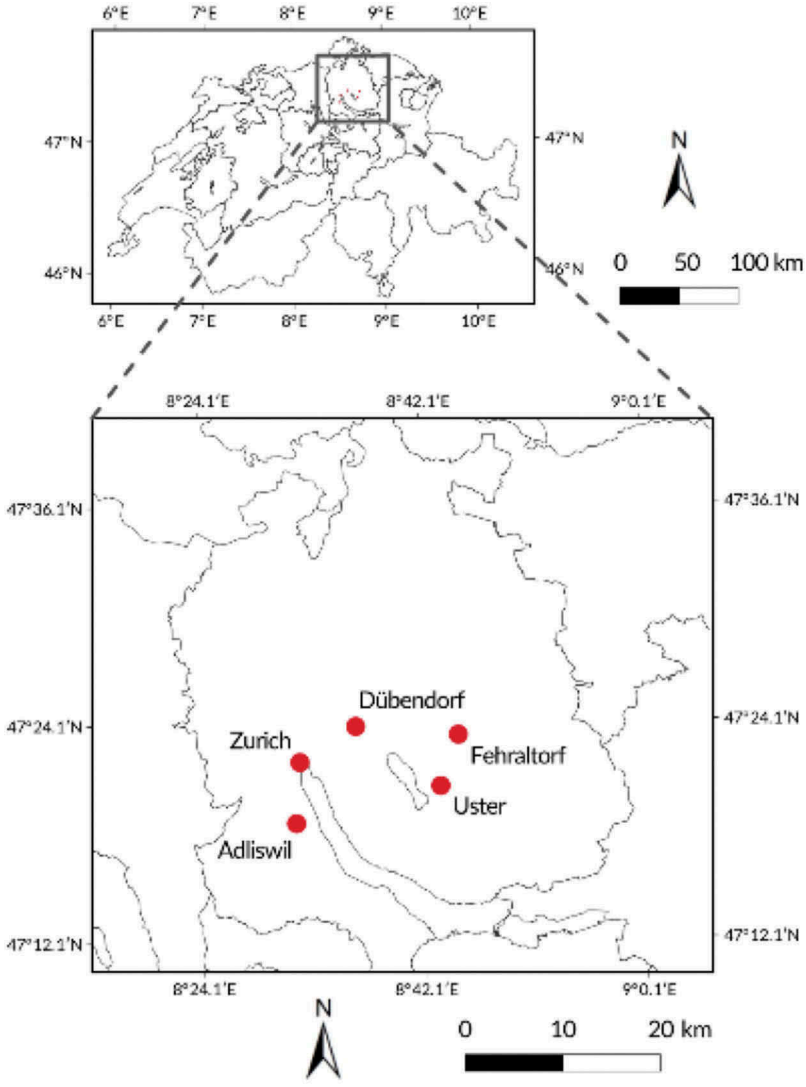


Figure 3. Areas of study used to test the framework.

performed with a backpropagation algorithm and stochastic gradient descent. In the case of object detection, the total loss to be minimized is the sum of the classification loss (i.e. deviation of predicted class from ground truth class) and the regression loss (i.e. deviation of predicted bounding box from ground truth bounding box).

The object detection model selected, a Faster R-CNN combined with Resnet-101, was implemented with the standard architecture without any further adjustments (full details on R-CNN and Resnet-101 are presented in Ren et al. (2017) and He et al. (2016), respectively) and trained using the Tensorflow Object Detection API[2]. The decision of selecting this model was based on the findings of Guo et al. (2016): *CNNs are the most utilized and most suitable deep learning techniques for images*, and, as previously described, on the results of the detailed deep learning models comparison conducted by Huang et al. (2017). To reduce the training time and the amount of required data, the internal model parameters were initialized with the parameters of a model pre-trained on the MS Coco dataset (Lin et al. 2014). This dataset contains 328k images with 2.5 million annotated common objects in natural settings.

The initialized object detection model was then trained with the annotated panoramas from the intra-regional training dataset. Before feeding the panoramas into the model, the panoramas and annotations were tiled (see Table B1 in Appendix B for chosen parameters). Furthermore, the data were augmented; tiles were randomly horizontally flipped to make the model less sensitive to whether an object is located on the images' left or right side. The training was terminated when the total loss, computed for the intra-regional validation dataset, converged after approximately 60 hours on a Nvidia Titan X Pascal 12GB GPU. Due to the long training time, the large number of hyperparameters and the (expected) marginal performance increase, no hyperparameters were optimized in this study. Nevertheless, a promising adjustment might be to add a smaller anchor[3] to the Faster R-CNN, complementing its three standard scales of 128 × 128, 256 × 256 and 512 × 512 px, which should improve the detection of small objects such as valves. In contrast, varying the anchor's ratio might be not as effective, as the used standard ratios of 1:1, 1:2 and 2:1 already cover the typical geometry of manholes, inlets and valves. In addition, due to the rather low number of objects per image, one could also decrease the set object proposals per image, which

potentially would lower the time for training and inference without any significant drop in object detection performance. The remaining hyperparameters (e.g. learning rate, batch size and number of epochs) can be tuned either manually or with automated procedures such as random search or gradient descent (Luo 2016).

4.4. Assessing the object detection performance

To assess the performance of the object detection model, it was fed with labelled tiles from the test datasets. For each tile, the model returned a set of bounding boxes and confidence scores for the detected objects in approximately 1 s (78 s for a full panorama image and approximately 24 hours for the whole study area). The confidence scores can be used to filter the detection results, e.g. to only retain detections that have a higher probability of being correct. The evaluation of the object detection performance was based on the criteria used in the Pascal VOC image recognition challenge (Everingham et al. 2010), which are widely accepted as standard quantitative evaluation measures in computer vision. To determine how well the detections match ground truth, the intersection over union (*IoU*) metric was used (Equation 9).

$$IoU = \frac{B_{detection} \cap B_{groundtruth}}{B_{detection} \cup B_{groundtruth}} \tag{9}$$

where $B_{detection}$ refers to the predicted bounding box while $B_{groundtruth}$ refers to the reference bound box are the respective bounding boxes of a detection and its corresponding ground truth (if available). Only detections with an *IoU* score equal to or larger than 0.5 and whose class match that of the ground truth were considered to be 'matches'. Matched detections are called true positives (*TP*) and the rest are false positives (*FP*). Ground truth bounding boxes that have not been matched with a detection are called false negatives (*FN*).

Using these definitions, summarizing metrics of precision and recall (Manning, Raghavan, and Schütze 2008) were considered. Precision is the fraction of detections that are true positives (Equation 10), while recall is the fraction of ground truth objects that were matched (Equation 11).

$$precision = \frac{TP}{TP + FP} \tag{10}$$

$$recall = \frac{TP}{TP + FN} \tag{11}$$

By varying the threshold applied to the detections' confidence scores, the relationship between precision and recall can be visualized in a so-called precision-recall curve for each class. The area below the precision recall curve is called the average precision (AP) and is commonly used to summarize the performance of an object detection model.

The F_1-score (Equation 12) is one of the metrics that can be used to assess the accuracy of a model. It is a function of precision (Equation 10) and recall (Equation 11) values, and is more robust to unbalanced class distribution than other performance metrics, e.g. the Accuracy metric.

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{12}$$

4.5. Assessing the object localization performance

Due to the magnitude of the errors encountered (in the order of several meters) and the absence of complete ground truth data for the urban drainage system elements (e.g. inventory map), the localization performance could not be reliably quantified. Instead, the localization performance was evaluated qualitatively with a high-resolution orthophoto in which the actual positions of the objects of interest can be deduced. The aerial images for the orthophoto were taken in April 2015 and May 2016, and are freely available from the online GIS system of the canton of Zurich[4]. The magnitude of localization errors as well as the lack of reliable reference data prevented an automatic and comprehensive quantitative evaluation of localization accuracy. Instead, locations in the data set were sampled pseudo-randomly and evaluated in a GIS software, in which the predicted object locations could be overlaid on high-resolution aerial images and compared.

5. Results

5.1. Object detection performance

5.1.1. Intra-regional test dataset

The precision recall curve obtained for the intra-regional test dataset is shown in Figure 4. Manhole covers showed the best performance with an average precision (AP) of 0.831 and average recall (AR) of 0.873, followed by drain inlets (AP = 0.786; AR = 0.829) (and valves (AP = 0.644; AR = 0.692)).This leads to relatively high F_1-score values of 0.851, 0.807 and 0.667 for manholes, drain inlets and valves, respectively. The differences in detection performance could be due to the typical size, complexity, and variety of the objects. Typical object location may also play a role, e.g. drain inlets are often found on the side of the road, whilst manhole covers are usually located in the centre of roads.

A representative selection of examples obtained from the test dataset is displayed in Figure 5. Figure 5(a) shows a typical scene with multiple objects of different classes. The model was able to detect less common drain inlet types (Figure 5(b)) or even objects under cars (Figure 5(c)). Furthermore, the model was

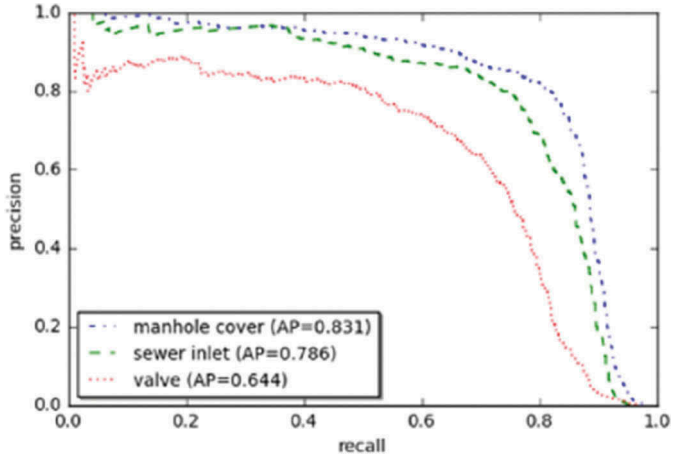


Figure 4. Precision recall curve for the intra-regional test dataset, containing 914 manhole covers and 669 drain inlets (and 484 valves). The IoU cutoff is set at 0.5.

successful even under extremely challenging illumination conditions such as bright light (Figure 5(e)) or shadows (Figure 5(f)).

There were also some cases that were difficult for the model to detect the objects. First, not all of the objects of interest were detected. These objects are often very challenging to detect, detectable only by an experienced human and after close examination. Second, the model also made false detections. For instance, cast iron grates for other purposes were also detected as manhole covers or drain inlets (Figure 5(d)). This example shows that the model lacks a deeper understanding of the relationship between interconnected parts, which is a known issue of the CNNs. To omit false detections, one could increase the threshold applied to the confidence score, since these detections tend to have a lower confidence score. However, some true positives will also be omitted in the process. The nonlinearity of this tradeoff is shown in the precision recall curve in Figure 4.

5.1.2. Extra-regional test dataset

The precision recall curve for the extra-regional test dataset is shown in Figure 6. The average precision achieved was 0.723 for manhole covers and 0.745 for drain inlets (valves show an average precision of 0.495). Compared to the intra-regional test dataset results, a significant drop in performance can be seen for all object classes. The AP obtained for manhole covers dropped by 0.108 (12.9%), and the AP of drain inlets decreased by 0.041 (6.3%). The AR and F_1-scores show a similar trend and are, respectively, 0.780 and 0.750 for manholes, 0.795 and 0.769 for drain inlets, and 0.560 and 0.525 for valves. This decrease in performance was expected, since peculiarities of the extra-regional dataset (e.g. lighting conditions, appearance of roads and infrastructure in Adliswil) were not necessarily represented in the intra-regional training data on which the model was trained.

5.2. Localization performance

Figure 7 provides an example that illustrates the common localization errors encountered in the results.

The situation shown in Figure 7 is representative of the magnitude and systematic nature of localization errors, which could not be quantified because of the lack of ground truth

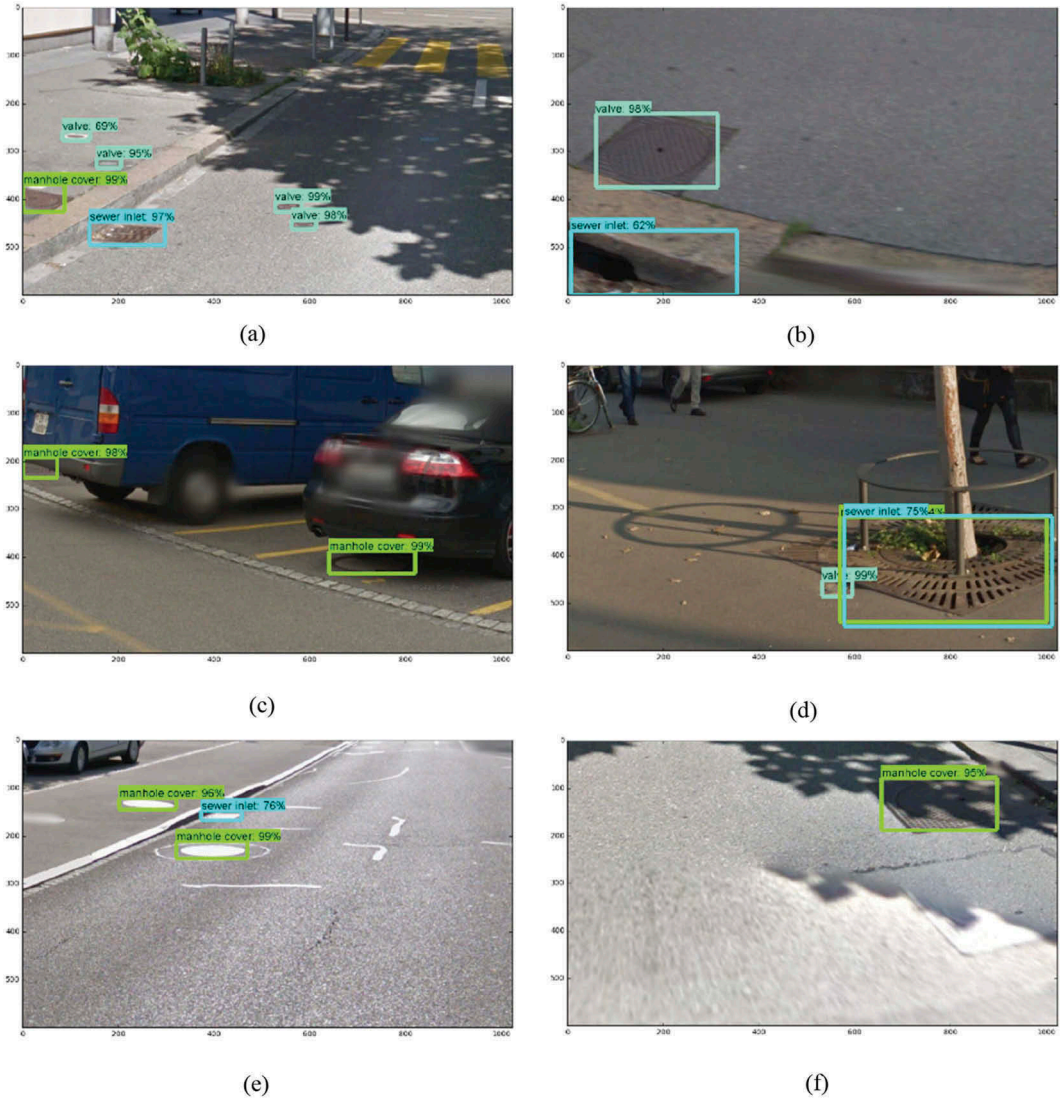


Figure 5. Examples of successful and unsuccessful object detections from the test dataset. (a) All three classes. (b) Less common drain inlet type. (c) Below a car. (d) False detection due to similar features. (e) Bright conditions. (f) Shady conditions.

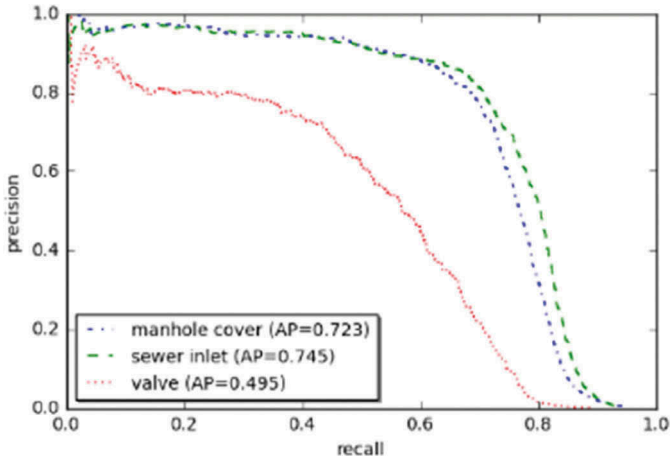


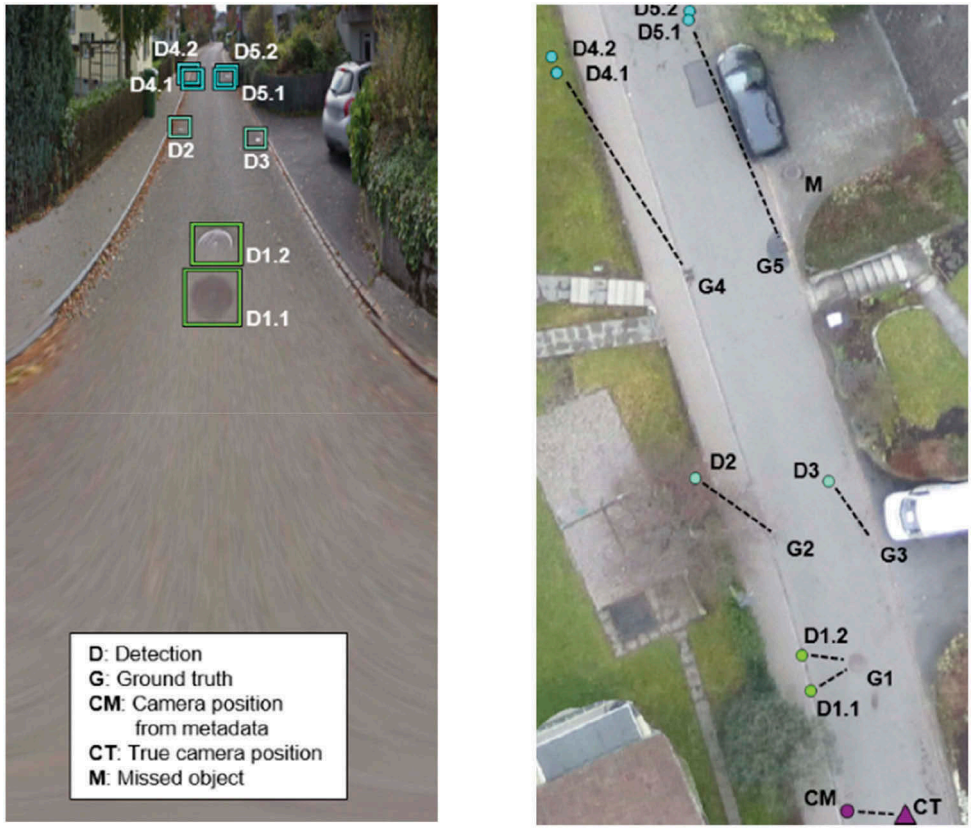
Figure 6. Precision recall curve for extra-regional test dataset, containing 2,264 manhole covers and 1,845 drain inlets (and 706 valves).

information and the magnitude of localization errors. The object closest to the camera is a manhole cover (G1) that is visible twice in the panorama due to stitching errors. The corresponding detections (D1.1 and D1.2) were estimated to be about two meters away from G1 (two valves (G2, G3) located about ten meters from the camera on each side of the road were also detected (D2, D3). Their predicted locations overshoot their actual locations by three to four meters). About 20 meters away from the camera, two drain inlets (G4, G5) were detected, but their estimated locations overshoot by about 10 meter (D4.1 & D4.2, D5.1 and D5.2). Both drain inlets were detected twice due to the tiling of the panorama. In this specific case, the issue of duplicate detections could be resolved by applying non-maximum suppression. One object (M) is not visible in this particular panorama and was therefore not detected. However, the missed object is visible in another panorama, where it is detected and localized. Overall, one can see that the localization error increases with the distance between the object and the camera, and an erroneous camera location adds error to the location of all the camera's detections.

6. Discussion

6.1. Comparison of object detection performance to previous work

In comparison to the approaches published to date, our approach achieved much higher object detection performance. The comparison with previous publications is made on the basis of the reported precision and recall, or average precision when available (which is a more comprehensive metric for comparison but not reported in all publications). It is important to note that since different datasets and different scenes were used for each approach, the conclusions that can be drawn from this comparison are limited.



(a) Street-level panorama (b) Aerial image

Figure 7. Example of a road in which infrastructure was detected. (a) Portion of one street-level panorama, with object detections (DX) represented by bounding boxes. (b) Aerial image of the same road section, including the estimated locations of the detections from (a) and indications of the manually labeled ground truth (GX).

For manhole cover detection in aerial images, Commandré et al. (2017b) reported a precision of 75% with a recall of 49%, whereas in the present study 80% precision and 70% recall (for the extra-regional test dataset) were obtained. However, the apparent performance gain of our approach should not be only attributable to the very high resolution data and the more advanced object detection model, but also a difference in how performance was measured: Commandré et al. (2017b) had access to a inventory database as a ground truth, whereas in this study manual image annotations to evaluate precision and recall were used, thereby disregarding obstructed objects in our evaluation. This may give an artificial advantage to our approach. In terms of computational cost, the approach of Commandré et al. uses a similar classifier but only analyzes one aerial image in contrast to multiple high-resolution street-level images. Because of this, their approach is expected to be faster than the current approach in most situations. With regards to drain inlets, Moy de Vitry et al. (2018) achieved an average precision of 0.73 thanks to a multiview approach with images taken from an unmanned aerial vehicle (UAV). In comparison, the average precision achieved in this study was 0.786 (for the intra-regional test results). Although Moy de Vitry et al. use a comparable amount of image data, the classification method is simpler, so computational effort is probably slightly below that of the present approach. Hebbalaguppe et al. (2017), who used Google Street View images to detect telecom infrastructure, report a precision of 54% with a recall of 47% for manhole detection. Despite the similar type of images, our approach still attains higher performance, probably thanks to the higher image resolution and more advanced detection model. Given that their approach uses the same tools as the present study, the computational cost should be similar, save for differences due to image resolution.

In summary, the street-level approach to urban drainage systems infrastructure detection appears to achieve best-in-class performance levels. Unfortunately, due to the different datasets and evaluation methods used in previous publications, it is not possible to fully determine which aspects of our approach most enhance detection performance. These factors, which could be investigated in the future, are (i) the high detail of street-level images, (ii) the object detection model, (iii) the inclusion of multiple similar classes in the annotations, (iv) the overlapping views of the street level images, and (v) the lower viewpoint in street level images.

6.2. Analysis of the localization error sources

Various sources of error could have contributed to the localization errors illustrated in Figure 7. First, the recorded camera position, collected via GPS, has some level of error. This can be seen by comparing the true camera position (CT), estimated from the panorama, and the recorded camera position (CM). The magnitude of the GPS error observed in this study is in the range of two to four meters, consistent with official reports on the accuracy of GPS measurements, which state a 95% confidence interval of 1.89 m for horizontal errors in single-frequency GPS receivers (WAAS T&E Team 2017).

The second possible source of error is the recorded heading yaw of the camera. This error component is virtually perpendicular to the camera's line of sight, and increases with the increasing planar distance of the object from the camera . For example, an error of 10° in the yaw results in a deviation of approx. 1.7 m at a distance of 10 m. Third, an error is introduced when the assumed height of the camera relative to the object is incorrect, which occurs if the camera height h in Equation 6 is incorrect or if the local road surface is not planar. Again, this error increases with the distance of the object from the camera. For example, at a distance of 10 m, an error of 0.5 m in the camera height results in a 1.7 m localization error. Fourth, inaccurate panorama stitching, inaccurate bounding boxes and the assumption that the object centroid is in the middle of the bounding box result in inaccurate specification of the object's centroid (x,y). These errors are expected to result in localization inaccuracies that are in the range of the object dimensions, and are therefore negligible when compared to the previously mentioned sources of error.

Finally, while the panoramas are assumed to have a cylindrical projection, this may not actually be the case, translating to a systematic error in the localization. Even if the assumption holds, errors can still be introduced during panorama stitching. Stitching errors occurred frequently in the panoramas used in this study, manifested as distortions and artefacts, such as that seen for the manhole in Figure 7(a).

6.3. Practical applications and future vision

Although the achieved detection performance can be considered good, the large localization errors observed currently restrict the application of our method to situations where precise geographic locations are required. For example, such information could be used to support urban flood modelling: Jang, Chang, and Chen (2018) showed that even when drain inlet locations are constrained to a 5 × 5 m grid, flood extents are still better estimated than when only manholes are used to route water between the surface and the drainage pipes. For other applications, such as inventory updates and drain inlet cleaning condition monitoring, higher localization performance is preferred, which involves addressing the sources of error listed in the previous section. Technically, it is already possible to produce street-level imagery with centimeter precise geolocation. This can be done with existing GPS technology such as Real-Time Kinematic (RTK) GPS, image registration, or assimilation of other sensor data such as LiDAR. Similarly, uncertainties in camera orientation and lens distortion are also relatively straightforward to resolve. Finally, the camera equipment and stitching software is continuously being improved, thereby providing better panorama quality.

7. Conclusions

In this work, it is shown that automatic analysis of street-level images to detect urban drainage systems infrastructure components is already feasible; the approach presented and demonstrated in this work can be implemented around the globe, wherever street-level images are available, as a first step to help tackle the current lack of information on urban drainage systems infrastructure. Filling this knowledge gap will contribute to more sustainable infrastructure management and improve the accuracy of urban flood models, used for urban flood risk assessment and management. Future work should focus on improving the process of localization, by, for example combining both street-level images

and high resolution imagery from unmanned aerial vehicles, and assessing the transferability and usability of the proposed approach.

Notes

1. https://developers.google.com/maps/documentation/streetview/intro.
2. https://github.com/tensorflow/models/tree/master/research/object_detection.
3. An *anchor* is a reference square box defined by its ratio and scale. For each image location (defined by a sliding window) the original Faster R-CNN implementation considers multiple anchors (nine in total: three scales and three ratios) (Ren et al. 2017).
4. https://maps.zh.ch/.

Disclosure statement

No potential conflict of interest was reported by the authors.

ORCID

João P. Leitão http://orcid.org/0000-0002-7371-0543

References

Agadakos, Y., K. Makantasis, P. Partsinevelos, G. Papadakis, and A. Doulamis 2013. "Safe Urban Growth: An Integrated ICT Solution for Unstandardized and Distributed Information Handling." In 2013 IEEE 14th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), Madrid, Spain, June 4–7. doi:10.1109/WoWMoM.2013.6583498

Barber, D. 2012. *Bayesian Reasoning and Machine Learning*. New York, NY: Cambridge University Press. ISBN: 978–0521518147.

Bartoli, O., N. Chahinian, A. Allard, J.-S. Bailly, K. Chancibault, F. Rodriguez, C. Salles, M.-G. Touroud, and C. Delenne 2015. "Manhole Cover Detection Using a Geometrical Filter on Very High Resolution Aerial and Satellite Images." In 2015 Urban Remote Sensing Event (JURSE), Lausanne, Switzerland. doi:10.1109/JURSE.2015.7120521

Branson, S., J. D. Wegner, D. Hall, N. Lang, K. Schindler, and P. Perona. 2018. "From Google Maps to a Fine-grained Catalog of Street Trees." *ISPRS Journal of Photogrammetry and Remote Sensing* 135: 13–30. doi:10.1016/j.isprsjprs.2017.11.008.

Chang, T.-J., C.-H. Wang, A. S. Chen, and S. Djordjević. 2018. "The Effect of Inclusion of Inlets in Dual Drainage Modelling." *Journal of Hydrology* 559: 541–555. doi:10.1016/j.jhydrol.2018.01.066.

Commandré, B., D. En-Nejjary, L. Pibre, M. Chaumont, C. Delenne, and N. Chahinian. 2017b. "Manhole Cover Localization in Aerial Images with a Deep Learning Approach." *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-1/W1: 333–338. doi:10.5194/isprs-archives-XLII-1-W1-333-2017.

Commandré, B., N. Chahinian, J.-S. Bailly, M. Chaumont, G. Subsol, F. Rodriguez, M. Derras, L. Deruelle, and C. Delenne 2017a. "Automatic Reconstruction of Urban Wastewater and Stormwater Networks Based on Uncertain Manhole Cover Locations." In 14th International Conference on Urban Drainage, Prague, Czech Republic

Everingham, M., L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. 2010. "The Pascal Visual Object Classes (VOC) Challenge." *International Journal of Computer Vision* 88 (2): 303–338. doi:10.1007/s11263-009-0275-4.

Guo, Y., Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew. 2016. "Deep Learning for Visual Understanding: A Review." *Neurocomputing* 187: 27–48. doi:10.1016/j.neucom.2015.09.116.

He, K., X. Zhang, S. Ren, and J. Sun 2016. "Deep Residual Learning for Image Recognition." In 2016 IEEE Conference on Computer Vision and Pattern Recognition, 770–778. doi:10.1109/CVPR.2016.90

Hebbalaguppe, R., G. Garg, E. Hassan, H. Ghosh, and A. Verma 2017. "Telecom Inventory Management via Object Recognition and Localisation on Google Street View Images." In 2017 IEEE Winter Conference on Applications of Computer Vision, 725–733. doi:10.1109/WACV.2017.86

Herold, M., M. Gardner, V. Noronha, and D. Roberts. 2003. Spectrometry and Hyperspectral Remote Sensing of Urban Road Infrastructure. *Journal of Space Communications* 3. Available at: https://spacejournal.ohio.edu/issue3/abst_herold.html (Accessed: 07 November 2019)

Hirabayashi, Y., R. Mahendran, S. Koirala, L. Konoshima, D. Yamazaki, S. Watanabe, H. Kim, and S. Kanae. 2013. "Global Flood Risk under Climate Change." *Nature Climate Change* 3: 816–821. doi:10.1038/nclimate1911.

Hoffmann, S., D. Hunkeler, and M. Maurer 2014. "Nachhaltige Wasserversorgung und Abwasserentsorgung in der Schweiz: Herausforderungen und Handlungsoptionen." *NFP 61 project report* http://www.nfp61.ch/SiteCollectionDocuments/nfp61_thematische_synthese_3_d.pdf]

Huang, J., V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fisher, et al. 2017. "speed/accuracy Trade-offs for Modern Convolutional Object Detectors." In 2017 IEEE Conference on Computer Vision and Pattern Recognition, 3296–3305. doi:10.1109/CVPR.2017.351

Hürter, H., and T. G. Schmitt 2016. "Die bunte Welt der Gefahrenkarten bei Starkregen – Ein Methodenvergleich." In Aqua Urbanica 2016, Rigi-Kaltbad, Switzerland, September 25–27.

Jang, J.-H., T.-H. Chang, and W.-B. Chen. 2018. "Effect of Inlet Modelling on Surface Drainage in Coupled Urban Flood Simulation." *Journal of Hydrology* 562: 168–180. doi:10.1016/j.jhydrol.2018.05.010.

Krylov, V. A., E. Kenny, and R. Dahyot. 2017. "Automatic Discovery and Geotagging of Objects from Street View Imagery." *arXiv* 1708.08417 [cs.CV]. doi:10.3390/rs10050661.

Leitão, J. P., N. E. Simões, R. D. Pina, S. Ochoa-Rodriguez, C. Onof, and A. Sá Marques. 2017. "Stochastic Evaluation of the Impact of Sewer Inlets' Hydraulic Capacity on Urban Pluvial Flooding." *Stochastic Environmental Research and Risk Assessment* 31 (8): 1907–1922. doi:10.1007/s00477-016-1283-x.

Lin, T. Y., M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick 2014. "Microsoft COCO: Common Objects in Context." In *Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science*, edited by D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, 8693. Springer: Cham. doi:10.1007/978-3-319-10602-1_48

Luo, G. 2016. "A Review of Automatic Selection Methods for Machine Learning Algorithms and Hyper-Parameter Values." *Network Modeling Analysis in Health Informatics and Bioinformatics* 5 (1): 18. doi:10.1007/s13721-016-0125-6.

Manning, C. D., P. Raghavan, and H. Schütze. 2008. *Introduction to Information Retrieval*. New York, NY: Cambridge University Press. ISBN: 978-0-521-86571-5.

Maurer, M., F. Chawla, J. von Horn, and P. Staufer. 2012. Abwasserentsorgung 2025 in der Schweiz. Dübendorf, Switzerland: Eawag.

Moy de Vitry, M., K. Schindlerm, J. Rieckermannm, and J. P. Leitãom. 2018. "Sewer Inlet Localization in UAV Image Clouds: Improving Performance with Multiview Detection." *Remote Sensing* 10: 706. doi:10.3390/rs10050706.

Niigaki, H., J. Shimamura, and M. Morimoto 2012. "Circular Object Detection Based on Separability and Uniformity of Feature Distributions Using Bhattacharyya Coefficient." In 21st International Conference on Pattern Recognition, 2009–2012, Tsukuba, Japan, November 11–15. doi:10.1094/PDIS-11-11-0999-PDN

Nirupama, N., and S. P. Simonovic. 2006. "Increase of Flood Risk Due to Urbanisation: A Canadian Example." *Natural Hazards* 40: 25–41. doi:10.1007/s11069-006-0003-0.

Pasquet, J., T. Desert, O. Bartoli, M. Chaumont, C. Delenne, G. Subsol, M. Derras, and N. Chaninian. 2016. "Detection of Manhole Covers in High-resolution Aerial Images of Urban Areas by Combining Two Methods." *Ieee Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 9 (5): 1802–1807. doi:10.1109/JSTARS.2015.2504401.

Ren, S., K. He, R. Girshick, and J. Sun, 2017. "Faster r-CNN: Towards Real-time Object Detection with Region Proposal Networks." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39 (6): 1137–1149. doi:10.1109/TPAMI.2016.2577031.

Seggelke, K., K.-H. Rosenwinkel, P. A. Vanrolleghem, and P. Krebs. 2005. "Integrated Operation of Sewer System and WWTP by Simulation-based

Control of the WWTP Inflow." *Water Science and Technology* 52 (5): 195–203. doi:10.2166/wst.2005.0134.

Sejnowski, T. J. 2018. *The Deep Learning Revolution*. Cambridge, MA: MIT Press. ISBN: 9780262038034.

Sun, C., A. Shrivastava, S. Singh, and A. Gupta 2017. "Revisiting Unreasonable Effectiveness of Data in Deep Learning Era." In IEEE International conference on computer vision, 843–852, Venice, Italy, October 22–29. doi:10.3389/fphys.2017.00843

Timofte, R., and L. van Gool 2011. "Multi-view Manhole Detection, Recognition, and 3D Localisation." In 2011 IEEE International Conference on Computer Vision Workshops, 188–195, Barcelona, Spain, November 6–13. doi:10.1109/ICCVW.2011.6130242

Viola, P., and M. Jones 2001. "Rapid Object Detection Using a Boosted Cascade of Simple Features." In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), 1:I – 511 – I – 518. IEEE Computer Society. doi: 10.1109/CVPR.2001.990517.

Voulodimos, A., N. Doulamis, A. Doulamis, and E. Protopapadakis. 2018. "Deep Learning for Computer Vision: A Brief Review." *Computational Intelligence and Neuroscience* 2018: 1–13. doi:10.1155/2018/7068349.

WAAS T&E Team. 2017. *GPS Performance Analysis Report #96*. Atlantic city, NJ. http://www.nstb.tc.faa.gov/reports/PAN96_0117.pdf

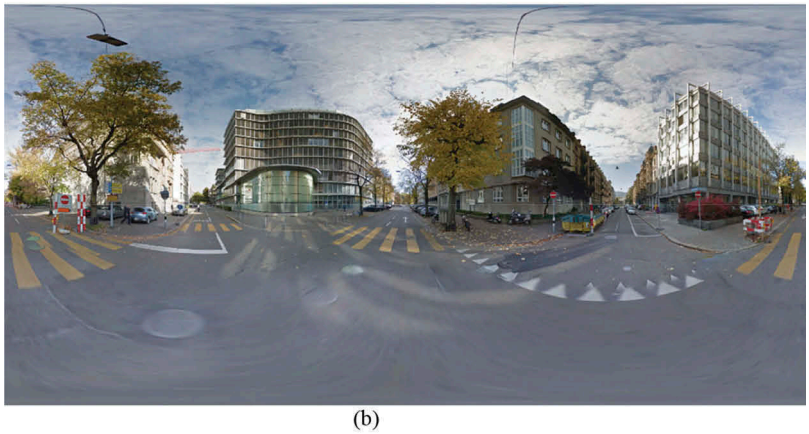
Wegner, J. D., S. Branson, D. Hall, K. Schindler, and P. Perona, 2016. "Cataloging Public Objects Using Aerial and Street-Level Images – Urban Trees." In 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, June 27–30. doi:10.1109/CVPR.2016.647.

Yu, Y., J. Li, H. Guan, C. Wang, and J. Yu. 2014. "Automated Detection of Road Manhole and Sewer Well Covers from Mobile LiDAR Point Clouds." *IEEE Geoscience and Remote Sensing* 11 (9): 1549–1553. doi:10.1109/LGRS.2014.2301195.

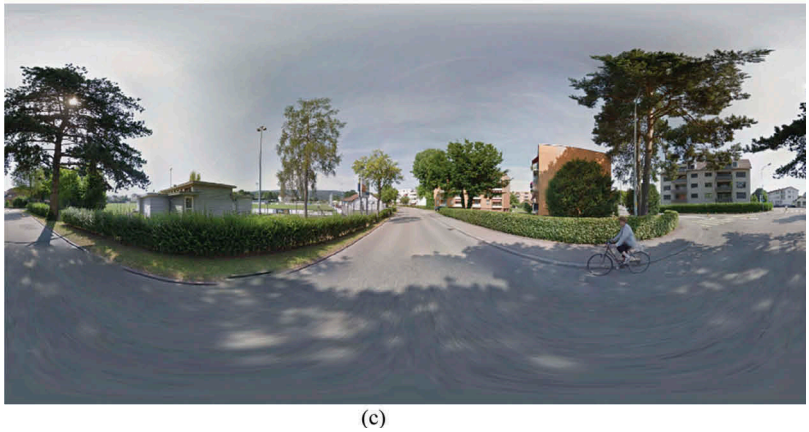
Yu, Y., J. Li, H. Guan, C. Wang, and J. Yu. 2015. "Automated Detection of Urban Road Manhole Covers Using Mobile Laser Scanning Data." *IEEE Transactions on Intelligent Transportation Systems* 16 (6): 3258–3269. doi:10.1109/TITS.2015.2413812.

Appendix A. Examples of Google Street View panoramas from the different municipalities (Zurich canton) used in this study

(a)



(b)



(c)

Figure A1. Examples of Google Street View panoramas for each selected municipality (Zurich canton, Switzerland). (a) Adliswil. (b) City of Zurich. (c) Dübendorf. (d) Fehraltorf. (e) Uster.

(d)

(e)

Figure A1. (Continued).

Appendix B. Parameters for the tiling step

Table B1. Parameters for tiling.

Parameter	Chosen value
Tile size	1,024 × 600 px
Tile's horizontal overlap	78 px
Tile's vertical overlap	78 px
Minimal width of tiled bounding box (β_{width})	10 px
Minimal height of tiled bounding box (β_{height})	10 px
Minimal area ratio (β_{area})	0.3