CCTV-FullyAware: toward end-to-end feasible privacy-enhancing and CCTV forensics applications

Hannu Turtiainen, Andrei Costin, Timo Hämäläinen, Tuomo Lahtinen, Lauri Sintonen
Faculty of Information Technology
University of Jyväskylä
Jyväskylä, Finland
{turthzu,ancostin,timoh,tutalaht,lamijosi}@jyu.fi

***Abstract*—It is estimated that over 1 billion Closed-Circuit Television (CCTV) cameras are operational worldwide. The advertised main benefits of CCTV cameras have always been the same; physical security, safety, and crime deterrence. The current scale and rate of deployment of CCTV cameras bring additional research and technical challenges for CCTV forensics as well, as for privacy enhancements.**

This paper presents the first end-to-end system for CCTV forensics and feasible privacy-enhancing applications such as exposure measurement, CCTV route recovery, CCTV-aware routing/navigation, and crowd-sourcing. For this, we developed and evaluated four complex and distinct modules (CCTVCV [1], OSRM-CCTV [2], BRIMA [3], CCTV-Exposure [4]), all of which are novel, unique, peer-reviewed, and can be used either separately or within an integrated end-to-end system such as CCTV-FullyAware. We release all our artefacts as open-source/open data. We hope our work will bootstrap policy-driving discussions and large-scale applications such as CCTV forensics and privacy-enhancing technologies.

***Index Terms*—machine learning, navigation, object detection, privacy-enhancing technologies, video surveillance**

I. Introduction

Closed-circuit television (CCTV) and video surveillance are everywhere nowadays. This technology has become one of the most ubiquitous in many cities, and it is nearly impossible to travel without encountering several cameras [5], [6]. CCTV cameras are a part of the modern city image; however, the earliest recorded knowledge of a forerunner of the modern CCTV camera is from 1927, when Leon Theremin installed "distance vision" cameras on Kremlin premises [7]. These mechanical devices transmitted "images" of enough quality to detect faces. From a cybersecurity perspective, CCTV cameras and other video surveillance systems are not without issues as they have been demonstrated to be susceptible to numerous cyberattacks [8]. They are also infamous as the main building blocks of the world-renown Mirai botnet [9], [10] due to the generally lax security of IoT devices [11], [12]. The privacy risks associated with CCTV cameras are also well established [13]–[17]. However, objectively assessing the privacy implications and risks of CCTV cameras is proving to be rather strenuous [18]. Moreover, CCTV forensics challenges [19] associated with the increasing number of CCTV cameras are increasing and are well-known.

Nevertheless, ways exist to mitigate the CCTV forensics challenges, the privacy risks, and the new technologies built into CCTV cameras. For example, low-tech and straightforward methods such as clear plastic masks [20] or makeup [21] are commonly used and promoted practices. However, as face recognition and other computer vision (CV) technologies advance, these methods cannot keep up with the arms race as face recognition has nowadays proven to be accurate even in detecting people wearing masks [22]. Other approaches could be presented as high-technology methods such as CCTV exposure and camera-aware navigation, as well as real-time systems to alert users via wearable or carry-with technologies of nearby cameras. These technologies require automation and Computer Vision (CV) for accurate object detection, mapping, localization, and counting cameras. These methods have been proven in research [23]–[25]. However, to our knowledge, such end-to-end CCTV-aware systems are non-existent as various fundamental building blocks are currently missing.

This paper showcases an end-to-end system that can assist with CCTV forensics (e.g. if a particular navigation route went nearby any CCTV cameras. This can help, in turn, to investigate criminal, humanitarian, and insurance cases by quickly identifying the subset of CCTV cameras that would be most interesting to capture evidence for the case), as well as with the privacy-preserving CCTV-aware routing and navigation system (e.g., to combat individuals' malicious and unwanted video recordings).

In this work, we explore the following applied research questions:

- RQ1) What are the necessary separate components that are both modular (i.e., can be used stand-alone to achieve their results and answer their research questions) but that are at the same time sufficient to be engineered together for a value-added end-to-end system for privacy and safety-enhancing applications working in a continuous feedback loop.
- RQ2) What are the design and engineering principles, steps, and challenges to achieving such an end-to-end system based on the independent modular components?

Our main contributions to this work are:

1) *First end-to-end system* aiming toward feasible CCTV forensics applications and privacy-enhancing CCTV-aware exposure, routing, navigation, and crowd-sourcing.
2) *Privacy-preserving CCTV-aware routing.* giving users

control over where they are being filmed and making conscious decisions to avoid these CCTV cameras.

3) *Mapping of CCTV cameras.* We developed a highly-technical system to map security cameras found on street view images to a navigation system map for privacy routing and CCTV forensics applications.
4) *Computer vision model to identify CCTV cameras from street view images.* Mapping cameras by hand would be an enormous task with no end. Therefore, an automated system is required, and we think using computer vision is the proper solution.
5) *Easy and fast annotation tool for CCTV cameras.* The training of our computer vision models requires a lot of image data. The data requires validation and annotation. Combining image gathering and annotation with an easy-to-use browser extension is logical to proceed with dataset gathering.

II. Related work

a) Routing, Navigation, and GPS-data: For open-source mapping and routing projects, *OpenStreetMap (OSM)* is a reputable choice. A lot of routing technology research [26]–[31] base their projects on the OSM, because of the open-source nature of it, free availability, and the editable maps feature. Routing solutions for wheel-chair accessibility [32]–[34] and day-arc based routes [35]–[37] are among the notable research. Commonly, routing criteria are based on route duration, length, and congestion. However, Siriaraya et al. [38] noticed alternative criteria for pedestrian routing based on qualities such as safety, exploration, and pleasure. We derived inspiration from previous day-arch routing research for our two-mode routing (privacy and safety). The gist of Olaverri Monreal et al. [35] routing solution was to avoid the sun. Similarly, Ma [36], [39] had two routing modes – facing or avoiding the sun. Even if it is sunny, trees can provide shade on one's travel, as Deilami et al. [37] noted in their research.

Safety and privacy routing modes are scratched upon in previous research. Hirozaku et al. [40] incorporated street light data into their routing solution, while Tessio et al. [41] set it to find greener and quieter streets for pedestrian routing. GPS data has been used for exposure modeling in many research papers. One of the most fruitful areas of research for this idea is the exposure to air pollutants, as it has been the subject of several papers [42]–[46]. GPS data and sensors were also used to measure soil radiation contamination in the Chornobyl exclusion area [47].

b) Object Detection: In our research, we utilized two premier object detection and classification frameworks – Detectron2 [48] from FAIR (Facebook AI Research) and MMDetection by K. Chen et al. [49]. Both frameworks are modular and flexible for training and utilizing state-of-the-art object detectors in single GPU systems and up top multi-node supercomputer clusters. They are also feature-rich and offer more straightforward tools on top of PyTorch and CUDA.

Our first model is based on ResNet split-attention networks by H. Zhang et al. [50]. The authors implemented novel split-attention blocks on top of the renowned Resnet backbone structure to enhance performance. The gist of the split-attention blocks is to divide feature maps more granularly and derive more representative features from these "splits" [50]. Our second model, based on DetectoRS by Qiao et al. [51], features a Recursive Feature Pyramid (RFP) and Switchable Atrous Convolution (SAC) mechanisms. RFP introduces feedback connections to achieve the "looking and thinking twice" arrangement. On the other hand, SAC improves the handling of objects of different scales [51].

Among the fast "real-time" object detectors, many versions of YOLO (You Only Look Once) have been arguably the most influential detectors. In 2020, Bochkovskiy et al. [52] proposed the fourth iteration of YOLO. They introduced several new features to the previous version 3 [53] and achieved 10% increase in average precision and 12% increase in frames-per-second in comparison. At the time, they stated state-of-the-art accuracy (MS COCO AP) and speed (FPS) with reasonable training system requirements, thus solidifying the anchor-based one-stage ("one look") methodology viability not only in fast real-time detection but in all-around object detection as well. As MS COCO [54] dataset features unlabeled (unannotated) data in addition to the labeled images, researchers have started to utilize this data with semi-supervised learning to improve their models' detection accuracy. At the time of writing, this model proposed by Xu et al. [55] is considered state-of-the-art in MS COCO dataset entries [56].

c) Computer Vision Datasets: Recently, general object detection competition has revolved around Microsoft's Common Objects in Context (COCO, or MS COCO). The latest update of MS COCO covers over 118,000 images with additional validation and testing datasets containing over 40,000 images. MS COCO consists of 80 different objects, such as traffic lights; however, CCTV cameras are missing from the list [57].

A famous general object detection challenge was held by PASCAL Visual Object Classes (VOC) [58] from 2005 to 2012 [59]. In 2012, the PASCAL VOC dataset had 20 object classes for their datasets, and they provided over 11,000 images containing over 27,000 object instances [60]. In order to increase the difficulty of the challenge, similar objects were chosen [61].

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [62] is an annual general object classification and detection challenge which started in 2010. The vision was to follow PASCAL VOC [58] and create a challenging competition. ImageNet contains over 1.2 million from training and over 150,000 images for validation and testing. ImageNet dataset has 1,000 classification classes; however, the authors initially chose 200 for challenge purposes [62].

Open Images Dataset V6 is a substantial general-purpose dataset containing over 9 million images with over 15 million annotated objects [63]. The project hosts an annual competition (Robust Vision Challenges) to maintain development in the object detection field [64].

The Mappilary Vistas work by Neuhold et al. [65] presented

a novel large-scale dataset built from about 25,000 street-level images. The images in the dataset were annotated into 66 object categories, while 37 object classes also have additional instance-specific annotations. However, this dataset contained less than 20 image instances of CCTV cameras, which is not enough to build a reliable and representative CV model out of it, and the authors did not attempt to build a CV model for CCTV camera detection, among other things. In comparison, our CCTVCV dataset offers manually reviewed and annotated $10,419$ CCTV camera instances. Recently, Sheng et al. [66] got inspired by our early CCTVCV work [1] and attempted to estimate the prevalence of surveillance cameras in 16 major cities. They built their CV models, which provided an accuracy rate of up to 93%, almost 6% less accurate than our CCTVCV model that we developed one year before their work. In a similar yet distinct direction, Buzzo [67] collected images of CCTV surveillance warning signs. To date, the author did not build a CV model out of it, and its main scope was an artistic-perspective exploration of surveillance in modern society.

d) Image Annotation: There are a plethora of image annotation tools available. Tools such as PhotoStuff [68] and LabelMe [69] are commonly used in research. Many of the annotation tools utilize annotation formats from large commonly used datasets (see Section II-0c) and from popular CV frameworks such as You Only Look Once (YOLO) [53]. LabelImg [70] is a Python-based annotation tool for PascalVOC [58] and YOLO [53] formats. Labelme by Wada [71] has taken inspiration from LabelImg; however, they added support for MS COCO [57] JSON format.

Machine learning and automation have also been utilized in annotation tools. For example, Ilastik [72], [73] is such a tool with image analysis capabilities, and ByLabel [74] brings semi-automatic labeling to the annotation field. Annotation tools have also been introduced to the portable device space as Wilhelm et al. [75] presented an annotation tool for mobile phones. For a more in-depth look at annotation tools, Hanbury [76] and Dasiopoulou et al. [77] provide in-depth surveys with an excellent choice of such tools.

e) Crowd-sourcing for Mapping and Image Annotation: As mapping and image annotation are laborious tasks on a large scale, crowd-sourcing is an obvious efficient way of managing the effort. With crowd-sourcing, the handlers can spread a simple task to many collaborators to cut down the amount of work for an individual. Su et al. [78] presented a system for bounding-box annotation via crowd-sourcing. However, to maintain quality and coverage, there are significant challenges that need to be addressed [76], [79]. One of such hurdles is the variance between annotator work quality and reliability [78]–[80].

III. System Concept

The challenge we envisioned was to create a system that would map CCTV cameras from street view images (both automatically and using manual crowd-sourcing) and then use that foundation for CCTV forensics and privacy-preserving routing options for navigation users. This high-level challenge had many nested challenges, which we treated as separate system components. In Fig. 1, we present our system concept.

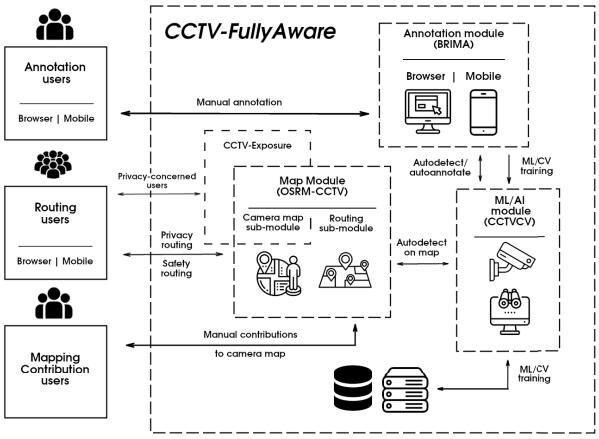

Fig. 1: Architecture of our end-to-end system for privacy-enhancing CCTV-aware routing, navigation, crowd-sourcing.

Our system operates on multiple concurrent processes to reach our end goal. We have three modules working together to serve three kinds of users and/or contributors. CCTVCV, our state-of-the-art CV object detection module (see Section IV-A), works with our annotation module BRIMA (see Section IV-B) to serve automatic detection of CCTV cameras for the annotation users but it also receives new training images from the annotation module. Our CV model also finds cameras from street view images fed through image-gathering algorithms (web-scrapers). We can capture geolocation data by parsing the URL using street view images. This addition helps us map cameras automatically and feed the data to our routing module (see Section IV-D). With the images and annotation data gathered from our annotation module, we can periodically retrain (transfer train) our CV model to further improve the results.

We have also developed a mobile-friendly version of our annotation tool for quick and easy camera mapping on the go. With Global Positioning System (GPS)-enabled devices, we can capture camera locations accurately.

Although our system is far from production-ready, we have already achieved encouraging preliminary results from our modules (see Section IV).

IV. System Components

In this section, we will elaborate on the findings of the individual system component papers. We will present the module and the preliminary results for each of them.

A. CCTVCV: Computer vision model for CCTV recognition

A vast database of camera locations and their attributes is required for the CCTV-aware routing option to be viable. Locating cameras manually, even with significant crowd-sourcing efforts, is not feasible on a large scale. Companies such

as Google and OpenStreetMap (OSM) produce and update street view mapping on a vast scale. Therefore, we argue that the most feasible and relatively accurate camera mapping is conducted with street view imagery. A CV object detector is required for this effort, for example, CCTVCV [1].

Our CV object detection model building consisted of four stages – dataset gathering with annotation, training environment setup, model training, and model testing. We ended up training and evaluating several models; however, for this paper, we only present the latest results [1].

Our best detectors were built using 8,387 images, which were manually reviewed and annotated to contain 10,419 CCTV camera instances, and achieved an accuracy rate of up to 98.7%. For a detailed description of the models themselves, we reference the interested readers to Section II-0b, where we extensively introduce existing works and their respective publications.

B. BRIMA: Image annotation tools for computer vision training data

Computer vision, image processing, and object detection research often require large datasets of images. As these efforts usually involve supervised learning, the objects in the images are needed to be annotated. In many cases, existing datasets are insufficient for the research; therefore, fast and easy tools are required for data gathering and annotation. Often, these tools can be cumbersome by offering many features, mandating an installation, and requiring a particular operating system. These issues can cause overhead on the research efforts. Image annotation tools are not scarce as multiple tools have been developed over the last decade [68]–[71], [81]–[83]. However, for our CCTV-aware project, we required our in-house built tool for our purposes and needs [3] as none of the existing tools allowed to capture annotations from "in-browser viewport" while navigating the Internet and street view pages.

Our BRowser-only Image Annotation tool (BRIMA) assists researchers by enabling a fast and easy way to create high-quality annotated image datasets for custom requirements. The annotation happens directly in the browser with our tool and thus does not require any installation. Our tool will work on any operating system with big-brand browser support. This feature enables the user to annotate any image on web pages and does not require additional pre-downloads. Our tool supports standard MS COCO [57] annotation format for compatibility with many common CV frameworks and pipelines. The predetermined requirements for our tool were; an easy User Interface (UI) and straightforward User Experience (UX) as well as a minimal setup as a browser add-on [3].

C. CCTV-Exposure: Measuring exposure to CCTV surveillance based on real-time and historical geo-location data and GPS tracks

The purpose of the CCTV-Exposure [4] module is to provide end-users with estimates to what degree the particular user has been exposed to CCTV surveillance based on the real-time and historical geolocation data from the user. We refer to the module as a "dosage meter" for CCTV. In essence, the module works as follows. The user either pro-actively supplies a GPS track file (e.g., GPS Exchange Format format [GPX]) or provides application access to their location history. The information is delivered to the CCTV-Exposure module for processing.

Technically our CCTV-Exposure module works as follows. The module searches through all the GPX points within the input data and cross-checks their distance to the cameras' location. For a more granular inspection of exposure, we interpolate more data points between the input data points to increase the accuracy of the results. In Fig. 2, we show a sample route through the city center. The blue points are data points from the GPX, and the red dots are camera locations. In Fig. 3, we highlight our data interpolation. Blue markers are GPX data points, and the red marker is a camera in the range of the points. The orange markers indicate interpolated points also in the camera range, which we calculated toward the previous and the next GPX points. We add these data to the exposure results. As both of the blue GPX data points in the example are within the range of the camera, we do not need to calculate interpolated points between them as the route between the points is a straight line, and the camera should, therefore, cover all of the distance.

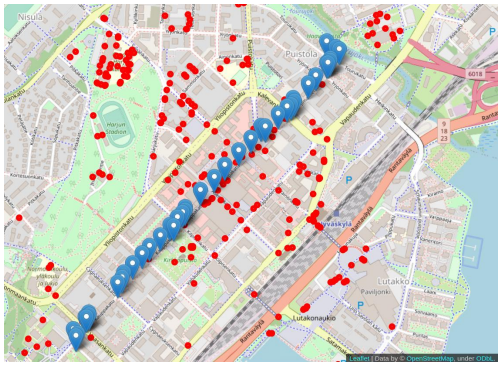

Fig. 2: Sample route in the city center of Jyväskylä [4]

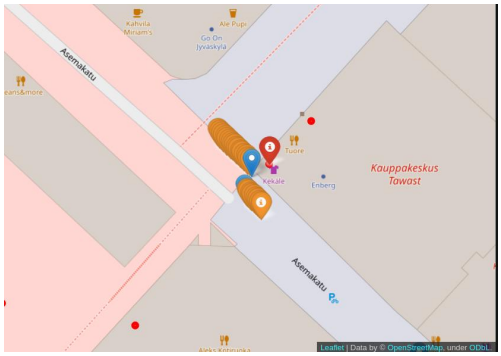

Fig. 3: Interpolated points for granular exposure calculation [4]

D. OSRM-CCTV: Open-source CCTV-aware routing and navigation system

Our routing setup is based on the *Open Source Routing Machine (OSRM)* [84], the profiles [85] built for it, and the traffic update [86] module. OSRM supports both privacy-first routing (i.e., route planning to avoid CCTV cameras on the route) and safety-first routing (i.e., route planning to route maximally where CCTV cameras are located). Inherently, the "privacy-first" and "safety-first" routing modes are just descriptive names, and we do not guarantee any actual privacy or safety. As detailed in OSRM-CCTV paper [87], we leave the large-scale quantitative and qualitative evaluation of both privacy-first and safety-first for immediate future work in a separate publication.

Our routing process is two-fold. First, an OSM file containing CCTV camera nodes, entrance and exit nodes of CCTV camera field-of-view, and OSM "ways" with a width property is created. For our routing capabilities, the width of the way is important as cameras can be avoided by, for example, walking across the street. Second, the processed file is loaded in Open Source Routing Machine (OSRM) [2], [84], [88].

For the routing backend, we run our *OSRM-CCTV* fork from the baseline Open Street Routing Machine (OSRM) [84], [88] project. We apply the standard backend [89] as features, such as blocking tags and traffic updates, are favorable for our project.

We identify over a dozen properties that affect the functionality and performance of the cameras. Many of these properties can be manually changed within the software. Cameras have different technical properties, settings, and recognition under the same conditions and can vary significantly [18]. Pixel Per Meter (PPM), or Pixel Per Foot (PPF), is a measure that can characterize the image detail a digital camera or imaging sensor can offer. Consequently, video surveillance tasks are assigned a minimum PPM for adequate operation [18]. European Standard EN 62676-4:2015 [90] depicts requirements for various video surveillance tasks. According to the researchers and the CCTV manufacturing companies, on average with surveillance grade cameras, people can be detected and tracked from 25 to 50 meters away, but face recognition requires a shorter distance of 15 to 20 meters [91], [92].

1) Results with real-world examples: Here we present a few real-world example on how our *privacy-mode* and *safety-mode* work on our *CCTV-aware* routing system in Jyväskylä, Finland, where we had manually mapped 450 CCTV cameras (see Fig. 4). All of the examples feature *privacy-mode* and *safety-mode* examples with the same start and end points [2].

In Figs. 5 and 6, we see our first example with exceptional results. The privacy route can avoid all the camera nodes and still arrive at the destination with privacy intact. In contrast, the safety route takes the shortest route while being in the cameras' field-of-view as much as possible [2].

The second example (Figs. 7 and 8) showcases how the privacy route (Fig. 7) needs to cross the street in order to avoid being recorded by a CCTV camera, while safety route (Fig. 8) stays in the spotlight the whole way [2].

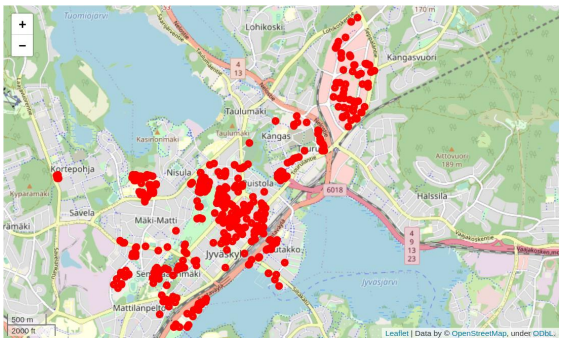

Fig. 4: Mapped cameras across the city center of Jyväskylä, Finland.

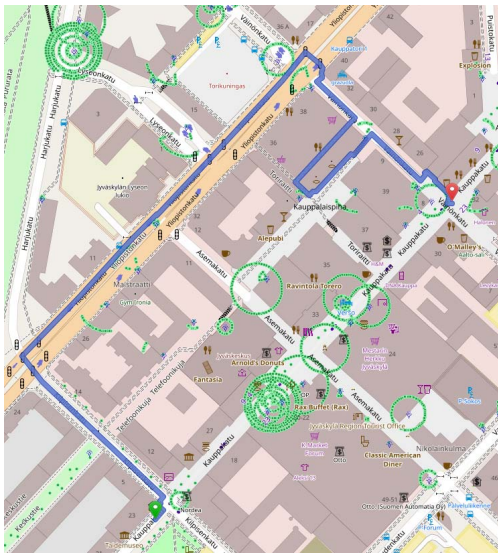

Fig. 5: *OSRM-CCTV* run with *privacy-mode* avoids all CCTV cameras in the downtown of Jyväskylä, Finland [2].)

V. System Workflow and Capabilities

As our end-to-end system has several interactive modules, we will present the system workflow based on the actions performed within the system.

a) Camera exposure: With our camera database and the CCTV-Exposure module, we can calculate the exposure distance and time for users who supply historical or on-demand data to the system. We call this procedure calculating the "CCTV dosage", which can help users further analyze their routes. For example, this can be used to analyze both future predictions and past retrospects, if a particular navigation route went nearby any CCTV cameras. This can help, in turn, investigate criminal, humanitarian, and insurance cases, e.g., by quickly identifying the subset of CCTV cameras that would be most interesting to capture evidence for the case.

b) Navigation and routing: Routing and navigation from an OSM-capable CCTV-aware navigation system are simple for the navigation user. As with any other standard routing option (i.e., walking, driving), the user selects either privacy

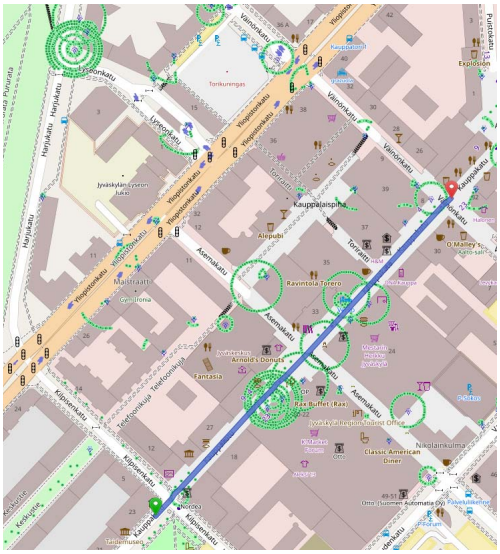

Fig. 6: *OSRM-CCTV* run with *safety-mode* routes straight through the fields of vision of multiple CCTV cameras in the downtown of Jyväskylä, Finland [2].

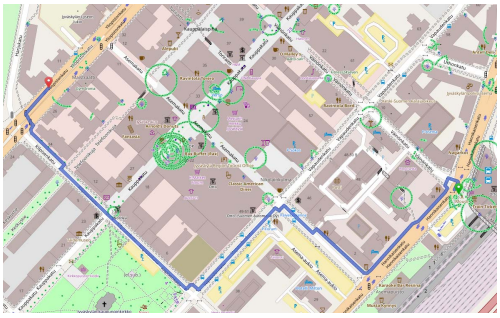

Fig. 7: *OSRM-CCTV* run with *privacy-mode* avoids all CCTV cameras in the downtown of Jyväskylä, Finland. Near the green marker on the right side of the image, *OSRM-CCTV* switches to the opposite side of the road to avoid nearby CCTV cameras [2].)

or safety -mode from the application and sets their route. Based on the user selection, the application chooses a routing server (or a local file) that corresponds to the selection, as the configuration file must be loaded beforehand. The routing engine performs the route in the range of feasibility.

c) Annotation: As our annotation tools work on a browser, there is no installation. The add-on is added to the existing browser, and the server configuration is set with a config file. The tool can work with any web page. However, if metadata collection is required, some setup is mandatory. Nevertheless, all of this setup can be done, for example, by a crowd-sourcing administrator; therefore, the users only need to use the supplied config file.

Users can capture an image with a “print-screen” button press when users have enabled the tool. When the capture is detected, annotation options come forth for the user. They can use segmentation to annotate their object, select appropriate options, and push the work to the server via the API.

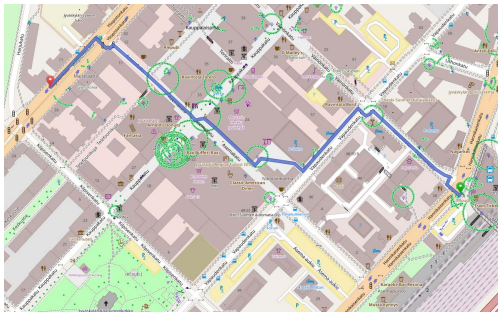

Fig. 8: *OSRM-CCTV* run with *safety-mode* chooses the its route to route through the fields of vision of multiple CCTV cameras in the downtown of Jyväskylä [2].

Bounding-box annotations are automatically calculated by the tool. The server saves the image and the annotations in MS COCO format.

d) Auto-detection annotation: For auto-detection-enabled annotation, the process is similar to standard annotation. However, the “autodetect”-button is now enabled in the UI. In this case, the image the user captured is sent to the backend for processing with the CV model. If model-defined objects are found, the annotations are sent back to the user for modification and approval. The user selects related meta-data from the supplied options and sends the image and annotation to the server like a typical annotation.

e) Mapping contribution: For mapping contribution users, the idea is to have users report cameras via the same OSM-capable navigation system they use for navigation and routing. The intent is to have a mode in the system where users can pinpoint a camera location, attach an image of it (street view or live image), and supply other camera metadata. Another camera mapping process could be automatic mappings via mobile devices. These features are yet to be fully integrated and tested at the time of writing, and the integration is left to future work.

f) Object detection model update: One of the core concepts of our end-to-end system is endless update capability. New data is gathered from automatic street view image scrapers or the annotation tool to upgrade the CV model. Data from the annotation tool is already in the correct format, and the data can be processed quickly and included in the training dataset for the next training session. Web-scraped images require “pseudo-labeling” (using our existing model to annotate the images) before adding them to any datasets. This operation can be set as a batch process that can be run, for example, as a certain amount of images is gathered.

For a future feature, we could utilize recent semi-supervised object detection model frameworks (such as presented in Section II-0b) to appropriately apply the “pseudo-labeled” annotations to the dataset for enhanced model progression.

g) Update camera locations to the mapping module: Once updated camera mapping has been gathered, the camera locations can be easily updated on a routing server where the system administrator needs only to replace the configuration

file. The configuration file can be generated with the server by supplying a new dataset of cameras and running our update scripts.

h) Challenges and disadvantages:

- *Cost and scalability challenges:* our system requires continuous crowd-sourcing efforts. Also, system monitoring by humans cannot be eliminated to ensure the proper functioning of the system.
- *Accuracy challenges:* our system requires continuous monitoring and improvement of the ML models. This process is resource intensive and can lead to data poisoning attacks (e.g., [93]).
- *Privacy challenges:* though the system works in fully anonymous mode, it requires users to upload potentially privacy-sensitive data that could, in theory, be used to deanonymize parts of the user identity (e.g., CCTV-Exposure GPS tracks).

VI. Conclusion

This paper presented the first end-to-end system aiming for feasible privacy-enhancing and CCTV forensics applications for CCTV-aware exposure, routing, navigation, and crowd-sourcing. For this, we developed and evaluated four complex and distinct modules (CCTVCV, OSRM-CCTV, BRIMA, CCTV-Exposure), all of which are novel and unique and can be used either separately or within an integrated end-to-end system. We leave the large-scale quantitative and qualitative evaluation of the CCTV-FullyAware end-to-end scenarios for immediate future work in a separate publication. We release all our artefacts as open-source/open data. We hope our work will bootstrap policy-driving discussions and large-scale applications, e.g., CCTV forensics and privacy-enhancing tech.

Acknowledgments

The authors acknowledge the grants of computer capacity from the Finnish Grid and Cloud Infrastructure (persistent identifier *urn:nbn:fi:research-infras-2016072533*). Part of this research was supported by a grant from the *Decision of the Research Dean on research funding within the Faculty (07.04.2021)*, and *Decision of the Research Dean on research funding within the Faculty (20.04.2022)* of the Faculty of Information Technology of University of Jyväskylä

Hannu Turtiainen also thanks the Finnish Cultural Foundation / Suomen Kulttuurirahasto (https://skr.fi/en) for supporting his Ph.D. dissertation work and research (under grant decision no.00221059) and the Faculty of Information Technology of the University of Jyvaskyla (JYU), in particular, Prof. Timo Hämäläinen, for partly supporting and supervising his Ph.D. work at JYU in 2021–2023.

The authors also acknowledge the use of royalty-free icons courtesy of www.flaticon.com (icons by: Good Ware, Freepik, itim2101, Pixel perfect, Icongeek26, Eucalyp, prettycons, and Stockio). Map image in Fig. 4 is generated with Folium (for Python) library (https://python-visualization.github.io/folium/) using OpenStreetMap data (https://www.openstreetmap.org).

Last but not least, the authors also thank Chairs of 2021 AAAI/ACM Conference on Artificial Intelligence, Ethics and Society (AIES) for their support and assistance solving a research ethics and integrity case related to our CCTVCV work, as detailed in [94].

References

[1] H. Turtiainen, A. Costin, T. Lahtinen, L. Sintonen, and T. Hamalainen, "Towards large-scale, automated, accurate detection of CCTV camera objects using computer vision. Applications and implications for privacy, safety, and cybersecurity.(Preprint)," 2020. 1, 3, 4

[2] L. Sintonen, H. Turtiainen, A. Costin, T. Hämäläinen, and T. Lahtinen, "OSRM-CCTV: CCTV-aware routing and navigation system for privacy and safety," in *12th International Conference on Business Modeling and Software Design (BMSD'22)*, 2022. 1, 5, 6

[3] T. Lahtinen, H. Turtiainen, and A. Costin, "BRIMA: low-overhead BRowser-only IMage Annotation tool," in *Proceedings: International Conference on Image Processing*. IEEE, 2021. 1, 4

[4] H. Turtiainen, A. Costin, and T. Hämäläinen, "CCTV-Exposure: System for measuring user's privacy exposure to CCTV cameras," in *12th International Conference on Business Modeling and Software Design (BMSD'22)*, 2022. 1, 4

[5] J. Pasley, "I documented every surveillance camera on my way to work in New York City, and it revealed a dystopian reality," https://businessinsider.com/how-many-security-cameras-in-new-york-city-2019-12, Dec 2019. 1

[6] D. Barrett, "One surveillance camera for every 11 people in Britain, says CCTV survey." 1

[7] A. Glinsky, *Theremin: ether music and espionage*. University of Illinois Press, 2000. 1

[8] A. Costin, "Security of CCTV and video surveillance systems: Threats, vulnerabilities, attacks, and mitigations," in *6th International Workshop on Trustworthy Embedded Devices (TrustED)*, 2016. 1

[9] M. Antonakakis et al., "Understanding the mirai botnet," in *26th USENIX Security Symposium (USENIX Security 17)*. USENIX Association, 2017. 1

[10] A. Costin and J. Zaddach, "IoT malware: Comprehensive survey, analysis framework and case studies," 2018. 1

[11] A. Costin, J. Zaddach, A. Francillon, and D. Balzarotti, "A large-scale analysis of the security of embedded firmwares," in *USENIX Security Symposium*, 2014. 1

[12] A. Costin, A. Zarras, and A. Francillon, "Automated dynamic firmware analysis at scale: a case study on embedded web interfaces," in *11th ACM on Asia Conference on Computer and Communications Security*, 2016. 1

[13] Electronic Frontier Foundation, "Street-Level Surveillance – Surveillance Cameras," https://eff.org/pages/surveillance-cameras. 1

[14] C. Slobogin, "Public privacy: camera surveillance of public places and the right to anonymity," p. 213, 2002. 1

[15] B. v. S.-T. Larsen, *Setting the watch: privacy and the ethics of CCTV surveillance*. Bloomsbury Publishing, 2011. 1

[16] J. Ryberg, "Privacy rights, crime prevention, cctv, and the life of mrs aremac," pp. 127–143, 2007. 1

[17] B. J. Goold, "Privacy rights and public spaces: Cctv and the problem of the "unobservable observer"," pp. 21–27, 2002. 1

[18] T. Lahtinen, L. Sintonen, H. Turtiainen, and A. Costin, "Towards CCTV-aware Routing and Navigation for Privacy, Anonymity, and Safety – Feasibility Study in Jyväskylä," in *Proceedings of Conference of Open Innovations Association FRUCT*, 2021. 1, 5

[19] R. Gomm, R. Brooks, K.-K. R. Choo, N.-A. Le-Khac, and K. W. Hew, "CCTV Forensics in the Big Data Era: Challenges and Approaches," *Cyber and Digital Forensic Investigations*, pp. 109–139, 2020. 1

[20] E. Heathcote, "Artists and activists offer privacy hope as facial recognition spreads," https://ft.com/content/15fb3c5a-2178-11ea-b8a1-584213ee7b2b, 2020. 1

[21] M. Lothian-McLean, "These activists use makeup to defy mass surveillance," https://i-d.vice.com/en_uk/article/jge5jg/dazzle-club-surveillance-activists-makeup-marches-london-interview. 1

[22] M. Pollard, "Even mask-wearers can be ID'd, China facial recognition firm says," https://reuters.com/article/us-health-coronavirus-facial-recognition/even-mask-wearers-can-be-idd-china-facial-recognition-firm-says-idUSKBN20W0WL. 1

[23] J. Fuentes-Pacheco, J. Ruiz-Ascencio, and J. M. Rendón-Mancha, "Visual simultaneous localization and mapping: a survey," pp. 55–81, 2015. 1

[24] D. Onoro-Rubio and R. J. López-Sastre, "Towards perspective-free object counting with deep learning," in *European Conference on Computer Vision*, 2016. 1

[25] G. Verhoeven, M. Doneus, C. Briese, and F. Vermeulen, "Mapping by matching: a computer vision-based approach to fast and accurate georeferencing of archaeological aerial photographs," pp. 2060–2070, 2012. 1

[26] OpenStreetMap, "Routing/online routers," https://wiki.openstreetmap.org/wiki/Routing/online_routers. 2

[27] ——, "Routing/offline routers," https://wiki.openstreetmap.org/wiki/Routing/offline_routers. 2

[28] O. Wiki, "List of osm-based services — openstreetmap wiki," 2020, [Online; accessed 1-November-2020]. [Online]. Available: https://wiki.openstreetmap.org/w/index.php?title=List_of_OSM-based_services&oldid=2052956 2

[29] D. Luxen and C. Vetter, "Real-time routing with OpenStreetMap data," in *19th ACM SIGSPATIAL international conference on advances in geographic information systems*, 2011. 2

[30] H. Bast, D. Delling, A. Goldberg, M. Müller-Hannemann, T. Pajor, P. Sanders, D. Wagner, and R. F. Werneck, "Route planning in transportation networks," in *Algorithm engineering*. Springer, 2016. 2

[31] R. J. Szczerba, P. Galkowski, I. S. Glicktein, and N. Ternullo, "Robust algorithm for real-time route planning," pp. 869–878, 2000. 2

[32] GIScience, Heidelberg Institute for Geoinformation Technology (HeiGIT), https://openrouteservice.org/, 2020. 2

[33] P. Kasemsuppakorn and H. A. Karimi, "Personalised routing for wheelchair navigation," pp. 24–54, 2009. [Online]. Available: https://doi.org/10.1080/17489720902837936 2

[34] A. Zipf, A. Mobasheri, A. Rousell, and S. Hahmann, "Crowdsourcing for individual needs—the case of routing and navigation for mobility-impaired persons," pp. 325–337, 2016. 2

[35] C. Olaverri Monreal, M. Pichler, G. Krizek, and S. Naumann, "Shadow as Route Quality Parameter in a Pedestrian-Tailored Mobile Application," 2016. 2

[36] K. Ma, "Parasol Navigation: Optimizing walking routes to keep you in the sun or shade," https://allnans.com/jekyll/update/2018/08/07/introducing-parasol.html, 2018. 2

[37] K. Deilami, J. Rudner, A. Butt, T. MacLeod, G. Williams, H. Romeijn, and M. Amati, "Allowing Users to Benefit from Tree Shading: Using a Smartphone App to Allow Adaptive Route Planning during Extreme Heat," p. 998, 2020. 2

[38] P. Siriaraya, Y. Wang, Y. Zhang, S. Wakamiya, P. Jeszenszky, Y. Kawai, and A. Jatowt, "Beyond the shortest route: A survey on quality-aware route navigation for pedestrians," *IEEE Access*, 07 2020. 2

[39] K. Ma, "Parasol: Shade model and routing algorithm for comfortable travel outdoors," https://github.com/keithfma/parasol. 2

[40] H. Miura, S. Takeshima, N. Matsuda, and H. Taki, "A study on navigation system for pedestrians based on street illuminations," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, 2011. 2

[41] T. Novack, Z. Wang, and A. Zipf, "A system for generating customized pleasant pedestrian routes based on openstreetmap data," 2018. 2

[42] J. Beekhuizen, H. Kromhout, A. Huss, and R. Vermeulen, "Performance of gps-devices for environmental exposure assessment," *Journal of exposure science & environmental epidemiology*, vol. 23, 2013. 2

[43] M. S. Breen, T. C. Long, B. D. Schultz, J. Crooks, M. Breen, J. E. Langstaff, K. K. Isaacs, Y.-M. Tan, R. W. Williams, Y. Cao *et al.*, "Gps-based microenvironment tracker (microtrac) model to estimate time–location of individuals for air pollution exposure assessments: Model evaluation in central north carolina," *Journal of exposure science & environmental epidemiology*, pp. 412–420, 2014. 2

[44] D. Dias and O. Tchepel, "Modelling of human exposure to air pollution in the urban environment: a gps-based approach," *Environmental Science and Pollution Research*, vol. 21, 2014. 2

[45] O. Tchepel, D. Dias, C. Costa, B. F. Santos, and J. P. Teixeira, "Modeling of human exposure to benzene in urban environments," *Journal of Toxicology and Environmental Health, Part A*, vol. 77, 2014. 2

[46] J. Ma, Y. Tao, M.-P. Kwan, and Y. Chai, "Assessing mobility-based real-time air pollution exposure in space and time using smart sensors and gps trajectories in beijing," *Annals of the American Association of Geographers*, vol. 110, no. 2, pp. 434–448, 2020. 2

[47] T. G. Hinton, M. E. Byrne, S. C. Webster, C. N. Love, D. Broggio, F. Trompier, D. Shamovich, S. Horloogin, S. L. Lance, J. Brown *et al.*, "Gps-coupled contaminant monitors on free-ranging chernobyl wolves challenge a fundamental assumption in exposure assessments," *Environment international*, 2019. 2

[48] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2: A PyTorch-based modular object detection library," https://ai.facebook.com/blog/-detectron2-a-pytorch-based-modular-object-detection-library-/. 2

[49] Kai Chen et al., "Mmdetection: Open mmlab detection toolbox and benchmark," 2019. 2

[50] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, Z. Zhang, H. Lin, Y. Sun, T. He, J. Mueller, R. Manmatha, M. Li, and A. Smola, "Resnest: Split-attention networks," 2020. 2

[51] S. Qiao, L.-C. Chen, and A. Yuille, "Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution," 2020. 2

[52] A. Bochkovskiy, C. Wang, and H. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *CoRR*, vol. abs/2004.10934, 2020. [Online]. Available: https://arxiv.org/abs/2004.10934 2

[53] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," 2018. 2, 3

[54] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," 2014. [Online]. Available: http://arxiv.org/abs/1405.0312 2

[55] M. Xu, Z. Zhang, H. Hu, J. Wang, L. Wang, F. Wei, X. Bai, and Z. Liu, "End-to-end semi-supervised object detection with soft teacher," *CoRR*, vol. abs/2106.09018, 2021. [Online]. Available: https://arxiv.org/abs/2106.09018 2

[56] "Papers With Code : Object Detection," https://paperswithcode.com/task/object-detection. 2

[57] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *European Conference on Computer Vision*. Springer, 2014. 2, 3, 4

[58] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes (VOC) challenge," 2010. 2, 3

[59] "The PASCAL Visual Object Classes," http://host.robots.ox.ac.uk/pascal/VOC/. 2

[60] M. Everingham and J. Winn, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Development Kit," 2012. 2

[61] L. Jiao, F. Zhang, F. Liu, S. Yang, L. Li, Z. Feng, and R. Qu, "A survey of deep learning-based object detection," 2019. [Online]. Available: http://dx.doi.org/10.1109/ACCESS.2019.2939201 2

[62] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," 2014. 2

[63] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov, T. Duerig, and V. Ferrari, "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale," 2020. 2

[64] "Open Images Dataset V6 + Extensions," https://storage.googleapis.com/openimages/web/index.html. 2

[65] G. Neuhold, T. Ollmann, S. Rota Bulo, and P. Kontschieder, "The mapillary vistas dataset for semantic understanding of street scenes," in *IEEE international conference on computer vision*, 2017. 2

[66] H. Sheng, K. Yao, and S. Goel, "CORRIGENDUM to "Surveilling Surveillance: Estimating the Prevalence of Surveillance Cameras with Street View Data" by Sheng et al." in *2021 AAAI/ACM Conference on AI, Ethics, and Society*, 2021, https://dl.acm.org/action/downloadSupplement?doi=10.1145%2F3461702.3462525&file=3462525-corrigendum.pdf. 3

[67] D. Buzzo, "Signs of Surveillance," in *Technology, Design and the Arts-Opportunities and Challenges*. Springer, Cham, 2020. 3

[68] W. Halaschek, J. Golbeck, A. Schain, M. Grove, B. Parsia, and J. Hendler, "PhotoStuff: An image annotation tool for the semantic web," in *Poster Track, 4th International Semantic Web Conference*, 2005, pp. 2–4. 3, 4

[69] B. Russell, A. Torralba, K. Murphy, and W. Freeman, "Labelme: A database and web-based tool for image annotation," pp. 157–173, 2008. [Online]. Available: http://dx.doi.org/10.1007/s11263-007-0090-8 3, 4

[70] Tzutalin, "LabelImg," 2015. 3, 4

[71] K. Wada, "labelme: Image Polygonal Annotation with Python," https://github.com/wkentaro/labelme, 2016. 3, 4

[72] S. Berg, D. Kutra, T. Kroeger, C. N. Straehle, B. X. Kausler, C. Haubold, M. Schiegg, J. Ales, T. Beier, M. Rudy *et al.*, "Ilastik: interactive machine learning for (bio) image analysis," pp. 1–7, 2019. 3

[73] C. Sommer, C. Straehle, U. Koethe, and F. A. Hamprecht, "Ilastik: Interactive learning and segmentation toolkit," in *2011 IEEE international symposium on biomedical imaging: From nano to macro*. IEEE, 2011, pp. 230–233. 3

[74] X. Qin, S. He, Z. Zhang, M. Dehghan, and M. Jagersand, "Bylabel: A boundary based semi-automatic image annotation tool," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 1804–1813. 3

[75] A. Wilhelm, Y. Takhteyev, R. Sarvas, N. Van House, and M. Davis, "Photo annotation on a camera phone," in *CHI'04 extended abstracts on Human factors in computing systems*, 2004. 3

[76] A. Hanbury, "A survey of methods for image annotation," pp. 617–627, 2008. 3

[77] S. Dasiopoulou, E. Giannakidou, G. Litos, P. Malasioti, and Y. Kompatsiaris, "A survey of semantic image and video annotation tools," in *Knowledge-driven multimedia information extraction and ontology evolution*, 2011. 3

[78] H. Su, J. Deng, and L. Fei-Fei, "Crowdsourcing annotations for visual object detection," in *Workshops at the 26th AAAI Conference on Artificial Intelligence*, 2012. 3

[79] P. Welinder and P. Perona, "Online crowdsourcing: rating annotators and obtaining cost-effective labels," in *Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*. IEEE, 2010. 3

[80] S. Nowak and S. Rüger, "How reliable are annotations via crowdsourcing: a study about inter-annotator agreement for multi-label image annotation," in *international conference on Multimedia information retrieval*, 2010, pp. 557–566. 3

[81] G. Ciocca, P. Napoletano, and R. Schettini, "IAT – Image Annotation Tool: Manual," 2015. 4

[82] J. Bernal *et al.*, "GTCreator: a flexible annotation tool for image-based datasets," pp. 191–201, 2019. 4

[83] F. Korc and D. Schneider, "Annotation tool," 2007. 4

[84] D. Luxen and C. Vetter, "Real-time routing with OpenStreetMap data," in *19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, ser. GIS '11. New York, NY, USA: ACM, 2011, pp. 513–516. [Online]. Available: http://doi.acm.org/10.1145/2093973.2094062 5

[85] Project-OSRM, "Osrm profiles," https://github.com/Project-OSRM/osrm-backend/blob/master/docs/profiles.md, 2020. 5

[86] ——, "Traffic," https://github.com/Project-OSRM/osrm-backend/wiki/Building-with-Mason, 2019. 5

[87] L. Sintonen, H. Turtiainen, A. Costin, T. Hamalainen, and T. Lahtinen, "OSRM-CCTV: Open-source CCTV-aware routing and navigation system for privacy, anonymity and safety (Preprint)," *arXiv preprint arXiv:2108.09369*, 2021. 5

[88] D. Luxen, "[OSM-dev] Announcing the immediate availability of the Open Source Routing Machine," https://lists.openstreetmap.org/pipermail/dev/2010-July/019834.html, 2010. 5

[89] D. Luxen and C. Vetter, "OSRM Backend," https://github.com/Project-OSRM/osrm-backend. 5

[90] B. EN, "62676-4: 2015. Video surveillance systems for use in security applications," *British Standard Institution*, 2015. 5

[91] F. W. Wheeler, R. L. Weiss, and P. H. Tu, "Face recognition at a distance system for surveillance applications," in *2010 Fourth IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*. IEEE, 2010. 5

[92] Axis, "Identification and recognition," https://axis.com/files/feature_articles/ar_id_and_recognition_53836_en_1309_lo.pdf. 5

[93] X. Chen, C. Liu, B. Li, K. Lu, and D. Song, "Targeted backdoor attacks on deep learning systems using data poisoning," *arXiv preprint arXiv:1712.05526*, 2017. 7

[94] H. Sheng, K. Yao, and S. Goel, "CORRIGENDUM to Surveilling Surveillance: Estimating the Prevalence of Surveillance Cameras with Street View Data," in *AAAI/ACM Conference on AI, Ethics, and Society*, ser. AIES '21. ACM, 2021. [Online]. Available: https://dl.acm.org/action/downloadSupplement?doi=10.1145%2F3461702.3462525&file=3462525-corrigendum.pdf 7