Mind your crossings: Mining GIS imagery for crosswalk localization

Dragan Ahmetovic,
Carnegie Mellon University

Roberto Manduchi,
University of California Santa Cruz

James M. Coughlan, and
The Smith-Kettlewell Eye Research Institute

Sergio Mascetti
Università degli Studi di Milano

Abstract

For blind travelers, finding crosswalks and remaining within their borders while traversing them is a crucial part of any trip involving street crossings. While standard Orientation & Mobility (O&M) techniques allow blind travelers to safely negotiate street crossings, additional information about crosswalks and other important features at intersections would be helpful in many situations, resulting in greater safety and/or comfort during independent travel. For instance, in planning a trip a blind pedestrian may wish to be informed of the presence of all marked crossings near a desired route.

We have conducted a survey of several O&M experts from the United States and Italy to determine the role that crosswalks play in travel by blind pedestrians. The results show stark differences between survey respondents from the U.S. compared with Italy: the former group emphasized the importance of following standard O&M techniques at all legal crossings (marked or unmarked), while the latter group strongly recommended crossing at marked crossings whenever possible. These contrasting opinions reflect differences in the traffic regulations of the two countries and highlight the diversity of needs that travelers in different regions may have.

To address the challenges faced by blind pedestrians in negotiating street crossings, we devised a computer vision-based technique that mines existing spatial image databases for discovery of zebra crosswalks in urban settings. Our algorithm first searches for zebra crosswalks in satellite images; all candidates thus found are validated against spatially registered Google Street View images. This cascaded approach enables fast and reliable discovery and localization of zebra crosswalks in large image datasets. While fully automatic, our algorithm can be improved by a final crowdsourcing validation. To this end, we developed a Pedestrian Crossing Human Validation (PCHV) web service, which supports crowdsourcing to rule out false positives and identify false negatives.

CCS Concepts

•Information systems → Geographic information systems; *Data extraction and integration*;*Crowdsourcing*; •Social and professional topics → People with disabilities; •Computing methodologies → Object detection; Epipolar geometry; •Human-centered computing → *Accessibility technologies*; Web-based interaction; Collaborative filtering

Additional Keywords and Phrases

Orientation and Mobility; Autonomous navigation; Visual impairments and blindness; Satellite and street-level imagery; Crowdsourcing

1. INTRODUCTION

Independent travel can be extremely challenging without sight. Many people who are blind learn (typically with the help of an Orientation and Mobility, or O&M, professional) the routes that they will traverse routinely [Wiener et al. 2010], for example to go to work, school or church. Far fewer attempt independent trips to new locations: for example to visit a new friend or meet a date at a restaurant. To reach an unfamiliar location, a person who is blind needs to learn the best route to the destination (which may require taking public transportation); needs to follow the route safely while being aware of his or her location at all times; and needs to adapt to contingencies, for example if a sidewalk is undergoing repair and is not accessible. Each one of these tasks has challenges of its own. In particular, the lack of visual access to landmarks (for example, the location and layout of a bus stop or the presence of a pedestrian traffic light at an intersection) complicates the wayfinding process. Thus, a straightforward walk for a sighted person could become a complex, disorienting, and potentially hazardous endeavor for a blind traveler.

Technological solutions for the support of blind wayfinding exist. Outdoors, where GPS can be relied upon for approximate self-localization, a person who is blind can use accessible navigation apps. While these apps cannot substitute for proper O&M training, they provide the traveler with relevant information on-the-go, or can be used to preview a route to be taken. A navigation tool, though, is only as good as the map it draws information from. Existing geographical information systems (GIS) lack many features that, while accessible by sight, are not available to a person who is blind. For example, Hara et al. found that knowing the detailed layout of a bus stop (e.g., the presence of features such as a bench or nearby trees) can be extremely useful for a person who is blind in figuring out where to wait for the bus [Hara et al. 2013a]. Other relevant information lacking in GIS may include the presence of curb ramps (curb cuts) near intersections, or the location of an accessible pedestrian signal controlled by a push button.

We propose a novel technique to detect zebra crossings on satellite and street level images. On four test areas, totaling $7.5 km^2$ in diverse urban environments, our approach achieves high quality detection, with precision ranging from 0.902 to 0.971, and a recall ranging from 0.772 to 0.950. Knowing the locations of marked crosswalks can be useful to travelers with visual impairments for navigation planning, finding pedestrian crosswalks and crossing

roads. A pedestrian, crossing a street outside of a marked crosswalk, has to yield the right-of-way to all vehicles[1]. While blind pedestrians do have right of way even in the absence of a crosswalk, we argue that marked crosswalks, being clearly visible by drivers [Fitzpatrick et al. 2010], and since they grant right-of-way to pedestrians, are a highly preferable location for street crossing in terms of safety. Pedestrians who are blind or visually impaired are taught sophisticated O&M strategies for orienting themselves to intersections and deciding when to cross, using audio and tactile cues and any remaining vision [Barlow et al. 2010]. However, there may be no non-visual cues available to indicate the presence and location of crosswalk markings.

Blind travelers may benefit from information about the location of marked crosswalks in two main ways. First, ensuring that a route includes street crossings only on clearly marked crosswalks may increase safety during a trip. Second, this information can be used jointly with other technology that supports safe street crossing. For example, recent research [Coughlan and Shen 2013; Mascetti et al. 2016; Ahmetovic et al. 2014] shows that computer vision-based smartphone apps can assist visually impaired pedestrians in finding and aligning properly to crosswalks. This approach would be greatly enhanced by the ability to ask a GIS whether a crosswalk is present, even before arriving at the intersection. If a crosswalk is present, the geometric information contained in the GIS can then be used [Fusco et al. 2014b] to help the user aim the camera towards the crosswalk, align to it and find other features of interest (e.g., walk lights and walk light push buttons).

This article extends our previous conference paper [Ahmetovic et al. 2015] with the addition of a survey of O&M professionals to better understand the role of pedestrian crossings in navigation for blind pedestrians (see Section 3). Additionally, we provide a new web service for validating the collected data through crowdsourcing by adding new crossings and filtering out false positives (see Section 4.4). A thorough evaluation of the performance of the automatic labeling system over different areas is presented in Section 5, along with a preliminary evaluation of the crowdsourcing system for label assignment refinement.

2. RELATED WORK

In recent years there has been an increasing interest in adding specific spatial information to existing GIS. Two main approaches can be identified: computer vision techniques and crowdsourcing.

Satellite and street-level imagery of urban areas in modern GIS are vast data sources. Computer vision methods can be used to extract geo-localized information about elements contained in these images (e.g., landmarks, vehicles, buildings). Satellite images have been used to detect areas of urban development and buildings [Sirmacek and Unsalan 2009], roads [Mattyus et al. 2015; Mokhtarzade and Zoej 2007] and vehicles [Leitloff et al. 2010]. [Senlet and Elgammal 2012] propose sidewalk detection that corrects occlusion errors by interpolating available visual data. In street level images, [Xiao and Quan 2009] propose detection of buildings, people and vehicles, while [Zamir and Shah 2010] tackle the issue of

---

[1]http://mutcd.fhwa.dot.gov

localizing user captured photographs. In a similar way, [Lin et al. 2015], localize street-level imagery in 45° aerial views of an area.

Recently, [Koester et al. 2016] proposed a technique based on Support Vector Machines (SVMs) to recognize pedestrian crossings from satellite images. The authors compare the performance of the proposed solution, in terms of precision and recall, with that reported in our previous paper [Ahmetovic et al. 2015] on the San Francisco dataset (SF1 region, see Section 5). While the solution proposed by Koester et al. has better performance than the first step of our classifier (which is based on satellite images only), our complete solution (i.e., based on satellite images and Street View panoramas) has superior performance. The previous solutions also present no way of addressing errors in the automated detection.

An alternative approach for detection of features in geo-spatial imagery is crowd-sourcing. In particular the impact of volunteered crowdsourcing, called volunteered geographic information (VGI), is stressed in [Goodchild 2007]. Wheelmap[2] allows accessibility issues to be marked on OpenStreetMap, the largest entirely crowdsourced GIS. [Hochmair et al. 2013] assess bicycle trail quality in OpenStreetMap while [Kubásek et al. 2013] propose a platform for reporting illegal dump sites.

Specific spatial information can be used to support people with visual impairments to navigate safely and independently. Smartphone apps that assist visually impaired individuals to navigate indoor and outdoor environments [Rituerto et al. 2016; Ahmetovic et al. 2016a; Ahmetovic et al. 2016b] have been proposed. For example, iMove[3] is a commercial application that informs the user about the current address and nearby points of interest. Large scale analysis of iMove usage data from 4055 users of the system shows how users can be clustered in different user groups based on their interest in different types of spatial information [Kacorri et al. 2016]. Other research prototypes, Crosswatch [Coughlan and Shen 2013] and ZebraLocalizer [Ahmetovic et al. 2011], allow pedestrians with visual impairments to detect crosswalks with a smart-phone camera. Similar solutions designed for wearable devices have the great advantage of not requiring the user to hold the device while walking [Poggi et al. 2015].

Crowdsourcing is also used to assist users with visual impairments during navigation. BlindSquare[4] is a navigation app that relies on the Foursquare social network for points of interest data and OpenStreetMap for street info. VizWiz[5] allows one to ask help of a remote assistant and attach a picture to the request. BeMyEyes[6] extends this approach to allow assistance through video feed. [Rice et al. 2012] gather information on temporary road accessibility issues (e.g., roadwork, potholes). StopInfo [Campbell et al. 2014] helps people with visual impairments to locate bus stops by gathering data about non-visual landmarks near bus stops. [Hara et al. 2013a] improve on this approach by performing crowdsourcing on street-level imagery, without the need to explore an area of interest in person. [Guy and

---

[2]http://wheelmap.org
[3]http://www.everywaretechnologies.com/apps/imove
[4]http://blindsquare.com/
[5]http://vizwiz.org
[6]http://www.bemyeyes.org

Truong 2012] propose an app to gather information on the structure and position of nearby crossings through crowdsourcing of street-level images and to assist users with visual impairments in crossing streets.

Computer vision techniques can use satellite and street-level images to assist people with visual impairments. [Hara et al. 2013c] propose the detection of inaccessible sidewalks in Google Street View images. [Murali and Coughlan 2013] and [Fusco et al. 2014a] match wide-angle (180–360°) panoramas captured by smartphone to satellite images of the surroundings for estimating the user's position in an intersection more precisely than with GPS.

Hybrid approaches using automated computer vision techniques for menial work and human "Turkers" for more complex tasks have also been proposed. [Hara et al. 2013b] extend the previous work to combine crowdsourcing and computer vision detection of accessibility problems, such as obstacles or damaged roads. In another work [Hara et al. 2014], a similar technique analyzes Street View imagery for the detection of curb ramps, which are then validated by crowd workers. Our approach, by using a satellite detection step followed by a Street View detection stage, limits the Street View validation to only those areas in satellite imagery in which candidates have been detected. Similarly, the open source project "OSM-Crosswalk-Detection: Deep learning based image recognition"[7] aims at recognizing pedestrian crossings from satellite images using a deep learning approach; results are validated by humans using the *MapRoulette*[8] software that is integrated in *OpenStreetMap*. A similar approach to crosswalk detection is proposed by the Swiss OpenStreetMap Association with another *MapRoulette* challenge aimed at validating crosswalks that have been automatically detected from satellite images[9].

To the best of our knowledge, our technique is the first to combine satellite and street-level imagery for automated detection, and crowdsourcing for validation of zebra crossings. The final crowdsourcing validation step makes it possible to add missed crosswalks, filter false detections, and can be extended to gather other information on the surroundings of the detected crossings, which can be of use to visually impaired pedestrians in finding and aligning to the crossings. Detected crosswalks can be added to a crowdsourced GIS and used by travelers with visual impairments during navigation planning. Solutions that detect crosswalks using the smartphone video camera [Ahmetovic et al. 2014; Coughlan and Shen 2013] can benefit from this information to assist the user in finding crossings at long distances that cannot be captured by the camera.

3. O&M SURVEY ON STREET CROSSING

While it may be intuitive for most sighted pedestrians that a marked crosswalk should be a safer place to cross a street than an unmarked one, one should be cautious when applying the same notion to the case of blind pedestrians. In order to achieve a clearer understanding of the importance of marked crosswalks for blind navigation, in this section we present a short

---

[7]https://github.com/geometalab/OSM-Crosswalk-Detection
[8]http://maproulette.org/
[9]http://sosm.ch/missing-crosswalks-a-maproulette-challenge/

survey that was administered to nine Orientation and Mobility (O&M) specialists. O&M professionals routinely help blind travelers in all aspects of urban navigation, from walking in a straight line by "shorelining" (maintaining constant distance to a surface parallel to the intended direction), to planning safe and efficient routes, to crossing a street. They represent an invaluable source of knowledge, as most of them have extensive experience with a wide variety of blind and visually impaired travelers.

3.1. Study Method

Since the rules of traffic can be quite different across countries, we decided to consult O&M specialists from the two different countries (the U.S. and Italy) that are home to the authors. Specifically, five respondents live in the U.S., while the remaining four are based in Italy. These individuals were recruited through the authors' network of acquaintances. While our sample is limited to only two countries, substantial differences already emerged from the survey, highlighting the risk of quick generalization and the importance of traffic norms on the way blind travelers plan their routes.

In order to understand the results of our survey, it is important to appreciate the rules regarding street crossing in the two countries considered. A crosswalk is defined as "the portion of a roadway designated for pedestrians to use in crossing the street" [Zegeer 1995]. In the U.S., crosswalks are implied at all road intersections, regardless of whether they are marked by painted lines or not [Kirschbaum et al. 2001]. In addition to intersections, mid-block crosswalks can be defined, but only if a marked crosswalk is provided. In contrast, pedestrians in Italy are supposed to only cross streets on marked crosswalks, overpasses or underpasses, unless these are located at more than 100 meters from the crossing point, in which case pedestrians can cross the street orthogonally to the sidewalk, giving the right of way to cars [Ancillotti and Carmagnini 2010]. To account for this difference, we worded our survey differently for the U.S. and the Italian versions. The survey questions are listed in the Appendix. Surveys were sent to the O&M specialists by email, giving them the choice to respond by email or via a telephone call. The U.S. respondents sent their responses by email, while the Italian ones preferred to use the phone.

3.2. Survey Results

The first two questions in the survey investigated the importance of choosing marked crosswalks in a route that requires crossing a street, even if this may imply a slightly longer route (Question 2). Somewhat surprisingly, the U.S. O&M specialists did not feel that blind pedestrians should preferably cross a street on a marked crosswalk. For example, one respondent said:

> Whereas it would be preferable for all legal crossings at intersections to be marked with high visibility crosswalk markings to help alert motorists of the potential presence of pedestrians, not all intersections are marked. The lack of crosswalk markings at an intersection does not make the crossing illegal or particularly unsafe.

Another respondent said:

> It might be "preferable" that they cross streets with painted lines, but it's not realistic. There are too many intersections without these lines, and there are too many negative consequences to address in finding an alternate intersection. Bottom line, they are not "unsafe" at crossings that do not have painted lines.

Likewise, all U.S. respondents, when faced with the hypothetical situation described in Question 2, said that they would advise taking Path 1 (which implies crossing a street at an intersection without a marked crosswalk) rather than Path 2 (which requires walking on a longer route in order to cross the street on a marked mid-block crosswalk). The answers to this question, however, were more nuanced. For example, one respondent pointed out that the deciding factor would not be the presence or absence of a marked crosswalk, but rather whether the crosswalk is controlled by a stop sign or a traffic signal; hence, unless the mid-block crosswalk were controlled, Path 1 would be preferable according to this O&M specialist. Another respondent said:

> One drawback in a mid-block crossing, even with a crosswalk, is that cars don't always anticipate someone crossing and may be going too quickly to stop easily or may just not be watching out for pedestrians as much as they do at corners.

A third respondent remarked that:

> Generally I teach Path 1 regardless of markings because of the ability to maintain a straight line of travel. Also its visibility is much greater in Path 1 allowing them to use traffic to cross safely. I avoid mid-block crossings as they are less safe due to restricted visibility and drivers generally not anticipating mid-block crossers.

Asked whether the availability of an app that helps the traveler locate the mid-block marked intersection would be a determining factor in the choice of Path 1 vs. Path 2, the U.S. respondents did not seem to think that it would. One commented:

> An app telling the traveler where the crosswalk markings are, in and of itself, would not be helpful (especially at an uncontrolled crossing). However, if that app could help them maintain a straight line of travel throughout the crossing (i.e. prevent them from veering), that would be of significant use.

Another respondent said:

> It would have to be a very savvy traveler to cross with technology [sic] as there is already an increased cognitive load with the additional challenges let alone diverting attention to the app.

A third one remarked:

> I would still not recommend wasting time and walking mid-block. Even with the app, it will still be somewhat challenging to determine where exactly to line up to make this mid-block crossing, especially in the absence of parallel traffic.

The Italian O&M specialists consulted in this survey gave a diametrically opposite response. All four respondents stated emphatically that a person who is blind should always try to cross a street on marked crosswalks, as long as they are identifiable by the pedestrian. For example, one respondent said (note: all quotations translated from Italian):

> Not all drivers recognize a white cane or a guide dog. Crossing the street on a zebra crosswalk is certainly preferable, and it is safer as the blind walker is protected by two rights: the right of a pedestrian [crossing on a marked crosswalk] and the right of a person who is blind [using a white cane or a guide dog].

They also generally opined favorably on Path 2 vs. Path 1, for multiple reasons: environmental and emotional control (*In Path 1, one is exposed to noise from multiple directions*, with *cars coming from all sides*), cultural factors (*We as drivers are used to give the right of way on marked crosswalks*), and the walker's own experience (*Crossing on the crosswalk is simpler*, although *the more clever ones can cross on the intersection*).

Question 3 in the survey asked for the type of information that a blind pedestrian would need when deciding where to cross a street, as well as the type of information that an existing or a hypothetical app could provide. Several of the respondents commented that the information requested by the first part of the question is basic O&M material (e.g. [Pogrund et al. 2012]), which any independent blind traveler should already be familiar with. This includes the ability to detect/identify corners/intersections, the ability to analyze any intersection they may come across, and strategies for lining up with the crossing. In fact, as one respondent pointed out:

> The "where" to cross (i.e. where to physically stand to line up for the crossing) is not as important as the "when" to cross (i.e. identifying the type of traffic control present).

None of the O&M specialists interviewed seemed to have much experience with apps such as Blindsquare, which give information about one's location and nearby streets. (In fact, one respondent commented that these apps sometimes "freeze up", leaving blind users unable to move on). Several respondents remarked that knowledge of whether the street is one-way or two-way, the presence of a traffic signal, as well as the name of the street and of nearby streets, would be very useful. One respondent remarked that an existing service, ClickAndGo[10], already provides this type of information. Two Italian respondents indicated that localization of the starting point of a zebra crossing would be useful. Other useful information that could be provided by a hypothetical app include: the width of the street to cross, the shape of the intersection, the volume of traffic, the visibility at a corner, and the traffic movement cycles. Several respondents remarked that a main challenge when crossing the street is locating the destination corner, given the difficulty of walking straight, especially in the absence of parallel traffic that can help define a path from departure to the destination curb.

The last question in our survey regarded the type of information that would be helpful to have in an intersection. We note in passing that at least one existing app, Intersection Explorer, available for Android, can provide some information about nearby street intersections. Since intersections automatically define crosswalks (at least in the U.S.), this question overlaps in part with Question 3. Respondents indicated that useful information would include: the cardinal directions of the streets in the intersection, along with the user's

---

[10]http://www.clickandgomaps.com

own location; the shape of the intersection (in particular, unusual layouts); the nearside parallel surge of the traffic flow; the presence of traffic controllers, including pedestrian acoustic signals; the presence of medians or of right turn islands; the presence of "do not cross" barriers; the type of traffic control (if present), and the location of pushbuttons for the activation of acoustic pedestrian signals; "anomalous" intersections containing diagonal crosswalks, or crosswalks that end against a wall or a vegetated area rather than on a sidewalk. One U.S. O&M specialist clearly stated that being informed of the presence or absence of crosswalks is not important; this contrasted with the opinion of two Italian respondents, who indicated that finding the locations of marked crosswalks is extremely important. Finally, two respondents mentioned the difficulties associated with crossing the street at a roundabout [Ashmead et al. 2005], a type of intersection layout that has become pervasive in Europe.

3.3. Discussion

In conclusion, this survey has highlighted some important aspects related to street crossing without sight, in particular *where* to cross a street, as well as the *information needs* of an app that could assist a blind traveler in these situations. Contrary to our expectation, the U.S. O&M respondents did not think that crossing a street at a marked crosswalk would be any safer than at an unmarked crosswalk. In contrast, the Italian respondents unanimously recommended crossing a street only at a marked crosswalk, if possible. This sharp difference in opinions is a consequence of the different traffic regulations in the two countries. We may conclude that prior knowledge of the location of marked crosswalks, as facilitated by the techniques discussed in this article, could be very important for blind pedestrians living in some but not all countries. In particular, the traffic code in the U.S. (and possibly in other countries) implies crosswalks at all intersections even without marking, which reduces the importance of using marked crosswalks for street crossing.

4. ZEBRA CROSSING IDENTIFICATION

Multiple types of marked crosswalks are used across the world. In the United States, at least two different types of marked crosswalks are available[11]. The *transverse marking* consists of two white lines, perpendicular to the road direction, with width between 6in (15cm) and 24in (60cm). The separation between the two lines is at least 6ft (180cm). *Zebra crossings*, known as "continental crossings" in USA, can be visually detected at larger distances than other marked crosswalks under the same illumination conditions [Fitzpatrick et al. 2010]. The zebra crossing pattern is often recommended because it is most visible to drivers [Knoblauch and Crigler 1987]. According to [Kirschbaum et al. 2001], "use of the continental design for crosswalk markings also improves crosswalk detection for people with low vision and cognitive impairments." In other countries (e.g., Italy), zebra crossing is the only form of marked crosswalk.

While in this contribution we focus on U.S. and Italian zebra crossings, the parameters of the detection can be tuned for other types of zebra crossings with different geometric

---

[11]http://mutcd.fhwa.dot.gov

characteristics. Other marked crosswalk types, such as transverse markings, could also be detected through appropriate computer vision techniques. We plan to extend our zebra-crossing detector to other transverse markings in future work.

A zebra crossing is a set of parallel, uniformly painted, white or yellow stripes on a dark background. The gaps separating the white stripes are of the same color of the underlying road and they define "dark stripes". Each stripe is a rectangle or, in case of diagonal crossings, a parallelogram. United States regulation dictates that zebra crossings be at least 6ft (180cm) wide, with white stripes 6in (15cm) to 24in (60cm) thick. The thickness of the dark stripes is not regulated. Italian regulation define zebra crossings as composed of at least 2 light stripes and 1 dark stripe. The stripes are 50cm thick and have a width of at least 250cm.

4.1. Technique Overview

Our system (Figure 1) is composed of two key components: an automated crossing detection cascade classifier and a manual crossing validation web service. The cascade classifier, depicted in Figure 2, first detects candidate zebra crossings from satellite images. These results are then validated on Google Street View panoramic images at the locations produced by the satellite image classifier. In addition to automated detection, we have created provisions for volunteers (Pedestrian Crossing Human Validation, or PCHV) to manually add crossings or delete false detections through an intuitive web interface (see Section 4.4). Finally, these results are stored in a database. We are currently storing this database on a private server, although we are planning to eventually port it to an accessible GIS system such as OpenStreetMap. External accessible apps can use the detected crossing information for navigation or route planning.

Given a certain region A of interest, one or more satellite images covering A are downloaded and analyzed by the algorithm described in Section 4.2. We have used Google Maps aerial images in our experiments, although of course other providers could also be used. We use a detector with a very high recall rate in this phase, since zebra crossings are never added in the subsequent stages (except possibly through human input in the parallel PCHV modality). Hence, a number of false positives should be expected. In the second stage, described in (Section 4.3), Google Street View panoramas (if available) are downloaded for locations close to the location of each satellite image detection. The purpose of this stage is to filter out any false detections from the satellite images. Of course, this may also result in the unfortunate removal of some zebra crossings correctly detected in the first stage.

The combined use of satellite images and Street View panoramas helps reduce the overall computational cost while ensuring good detection performance. While Street View images have much higher resolution than satellite images (and thus enable better accuracy), downloading and processing a large number of panoramas would be unwieldy. For example, in experiments with our SF1 dataset (Section 5), we found that surveying an area A of 1.6km^2 would imply downloading approximately 637MB of panoramic image data. The amount of satellite image data for the same area is much smaller ($\approx$ 23MB), with only 16MB of panoramic image data required for validation. Thus, in this example, our cascaded

classifier only needs to download and process 6% of the data that would be required by a system that only analyzes Street View panoramas.

4.2. Satellite Image Processing

Algorithm 1 describes our procedure to acquire and process the satellite images[12]. We rely on the *Google Static Maps API*[13] to download satellite images through HTTP calls. Each HTTP call specifies the GPS coordinates of the image center, the zoom factor, and the image resolution. The maximum image size for downloading is constrained to 640×640 pixels; hence, for large input areas, several HTTP requests need to be sent. Note that in the following we use the maximum zoom factor, at which the image size corresponds to an area of approximately 38m×38m.

ALGORITHM 1

Satellite images acquisition and processing

Input: Rectangular geographical area *A*.

Output: a set *Z* of zebra crossings, each one represented by its position and direction.

Method:

1:	$Z \leftarrow \emptyset$ *{algorithm result}*
2:	partition *A* into a set *R* of sub-regions
3:	**for all** (sub-region $r \in R$) **do**
4:	**if** (*r* does not contain a road) **then continue**
5:	download satellite image *i* of area *r*
6:	generate extended image *i′*
7:	$L \leftarrow$ detect line segments in *i′*
8:	$S \leftarrow$ group line segments in *L* in candidate crossings
9:	**for all** (candidate crossing $s \in S$) **do**
10:	**if** (*s* is not a valid crossing) **then continue**
11:	$z \leftarrow$ position and direction of zebra crossing *s*
12:	merge and add *z* to *Z*
13:	**end for**
14:	**end for**
15:	**return** *Z*

A region of interest *A* is acquired by downloading enough satellite images to cover it (Figure 3(a)). The locations and areas of these images (*sub-regions*) are easily computed given the zoom factor. In fact, not all sub-regions need to be downloaded, but only those that contain one or more roads (since a region without a road cannot contain a zebra crossing). This information can be obtained through the *Google Maps Javascript API*[14], which exposes a method to compute the closest position on a road to a given location. Thus, before downloading an image centered at a point *p*, we compute the distance from *p* to the closest road; if this distance is larger than half the diagonal length of the image, we conclude that

---

[12]Satellite and Street View imagery courtesy of Google ©
[13]https://developers.google.com/maps/documentation/staticmaps/
[14]https://developers.google.com/maps/documentation/javascript/

the image does not include any roads, and avoid downloading it (Algorithm 1, Line 4). Besides reducing the overall number of images to be downloaded (note that there is a maximum daily number of requests that can be submitted to Google Maps), this simple strategy helps reducing the rate of false positives.

Dividing A in subregions implies that a zebra crossing spanning multiple adjacent images may be recognized in some of them or may not be recognized in any. Consider the example in Figure 3(c), where the white dashed line is the boundary between two adjacent images. To tackle this problem, we construct an *extended image* by merging a downloaded image with the borders of the 8 surrounding images, as shown in Figure 3(b) (see Algorithm 1, Line 6). The width of each border is chosen so as to guarantee the recognition of a zebra crossing, even if it lies on the edge between adjacent images. Still, this approach may result in a crossing being recognized in more than one image, a situation that we need to specifically address.

Our strategy for zebra crossing detection is adapted from the ZebraLocalizer algorithm for zebra crossing recognition on a smartphone [Ahmetovic et al. 2014]. First, we find line segments corresponding to the long edges of stripes (Figure 4(b)); these are detected using a customized version of the EDLines algorithm [Akinlar and Topal 2011] (Line 7). The line segments thus obtained are grouped into sets of stripes based on criteria that include horizontal distance, vertical distance, and parallelism (Line 8). The resulting candidate crossings (Figure 4(d)) are validated or discarded based on the number of stripes they contain and on their grayscale intensity (Line 10). Note that, whereas the ZebraLocalizer requires reconstruction of the ground plane to remove perspective distortion [Ahmetovic et al. 2014], this operation is not needed in this case, as satellite images can be assumed to be frontoparallel (with the ground plane parallel to the image plane).

A detected zebra crossing is characterized by its orientation (specifically, the orientation of a line orthogonal to the parallel stripes) and by its position (represented by the smallest quadrilateral encompassing the detected set of stripes, Figure 4(d)). Each crossing is added to the set of results Z. If Z already contains a zebra crossing with approximately the same position and direction (which could happen when, as discussed earlier, the same crossing is viewed by multiple regions), the two crossings are merged into one.

4.3. Street View Image Processing

Algorithm 2 describes our procedure for validating a single zebra crossing, detected from satellite imagery, by means of Street View panoramas. Note that this procedure is repeated for all zebra crossings detected on satellite images.

In *Google Maps*, Street View panoramas are spherical images (i.e., they span 360° horizontally and 180° vertically) positioned at discrete coordinates, distributed non-uniformly in space. These panoramas are structured as a graph that closely follows the road graph. The panorama closest to a given location is retrieved through the *Google Maps Javascript API*; this also returns the exact location of the panorama.

ALGORITHM 2

Street View images acquisition and processing

Input: candidate zebra crossing $z \in Z$ represented by its position and direction.

Output: a validated zebra crossing z' represented by its position and direction or **null** if not validated.

Method:

1:	$c_0 \leftarrow$ get the coordinates of panorama closest to z
2:	$C \leftarrow \{c_0\}$ {Set of panoramas to be processed}
3:	**while** ($C \neq \emptyset$) **do**
4:	$c \leftarrow$ pop element from C
5:	$a \leftarrow$ direction angle from c to z
6:	$i \leftarrow$ image at coordinates c with direction $alpha$
7:	$L \leftarrow$ detect line segments in i
8:	$L' \leftarrow$ rectify line segments in L
9:	$S \leftarrow$ group line segments in L' in candidate crossings
10:	**for all** ($s \in S$) **do**
11:	**if** (s is a valid crossing) **then**
12:	$z' \leftarrow$ position and direction of zebra crossing s
13:	**if** (z matches z') **then return** z'
14:	**end if**
15:	**end for**
16:	push in C coordinates of panoramas directly linked to c and close to z
17:	**end while**
18:	**return null**

Note that the closest panorama to a true zebra crossing location is not guaranteed to provide a good image of the crossing. This could be due to occlusion by the vehicle used to take the pictures, or by other objects and vehicles. For example, Figure 5 shows in its center a crossing detected in a satellite image. This crossing is not visible in the closest panorama, shown to the right, due to occlusion by a car. The crossing, however, is highly visible in another panorama, shown to the left. Hence, in order to increase the likelihood of obtaining a panorama with the zebra crossing clearly visible in it, we download multiple panoramas within a certain radius around the location determined by satellite image analysis. These panoramas are analyzed in turn, beginning with the closest one, until a crossing is detected. If no crossing is found in any of the downloaded panoramas, the candidate detection from the satellite is rejected.

Only a small portion of a spherical panorama needs to be analyzed for the purpose of zebra crossing validation. Since we know the location z of the center of the crossing as detected in the satellite image, as well as the location of the camera that took the panorama, we can identify the portion of the panorama facing z within a horizontal field of view of 60° (see Figure 6(b)). The vertical pitch is kept at (−30°) with vertical field of view (60°) (shown in Figure 6(a)); this allows us to exclude from analysis the portion of the panorama that contains the image of the vehicle used to take the images and the entire portion of the image

above the horizon. Note that, while these fixed viewing parameters have given good results in our experiments, more complex strategies could be devised to adapt the algorithm to the relative position of the camera vs. the detected zebra crossing location z.

The image rendered from a panorama with the viewing geometry described above is processed using an adapted version of ZebraLocalizer [Ahmetovic et al. 2014] algorithm to reconstruct the coordinates of line segments visible on the ground plane (Algorithm 2, Lines 7 to 11). Note that, since the position and orientation of the virtual camera embodying this viewing geometry is known, the geometric transformation (homography) from the ground plane to the image plane is easily reconstructed. This allows us to warp (rectify) the image to mimic the view from a camera placed fronto-parallel with the ground plane, thus facilitating detection of the zebra crossing. The rectified line segments are then grouped and validated based on the same criteria described in Section 4.2.

If a zebra crossing is detected in the panoramic image, it position and orientation are computed and compared with those of the crossing detected in the corresponding satellite image. A certain amount of misregistration is to be expected; for example, the same crossing detected in a satellite image (Figure 7(a)) and in a Street View panorama (Figure 7(b)) may have slightly different GPS coordinates (Figure 7(d)). Thus, any comparison between the two crossing detections must allow for some tolerance in both position and orientation.

4.4. Validation via Crowdsourcing

Although our automatic zebra crossing detector produces good results (Section 5), missed detection and false positives are to be expected, due to multiple reasons (including poor image quality, occlusions and missed views of a zebra crossing in the Street View panoramas). In order to increase the accuracy of these results, we created a simple web interface mechanism that allows human volunteers to add or remove data from the database populated by our system (using an approach similar to that of Hara et al. [Hara et al. 2013c]). Volunteers are able to access the same data available to the automatic system (satellite and Street View images). This service, called Pedestrian Crossing Human Validation or PCHV, is publicly available[15].

Figure 8 shows a screenshot of a web page from PCHV. The interface is divided in two panes. The satellite image is displayed in the left pane, with all detected crossings in the database shown as red pins. The user may add new pins by clicking on the map, or may drag existing pins to change their location. Double-clicking on a pin removes the associated crossing. A single click on a pin triggers display of the location of nearby panoramas, shown as grey pins. The user can click on any such pins to visualize the associated panorama in the left pane. Note that, as mentioned earlier, evaluation of multiple panoramas may be necessary in bad visibility conditions.

---

[15]http://webmind.di.unimi.it/satzebra/pchv.html

5. EXPERIMENTAL EVALUATION

This section reports the results of the experiments conducted to evaluate our technique These results are quantified in terms of precision (the fraction of correctly detected zebra crossings over all detected crossings) and recall (the fraction of correctly detected zebra crossings over all the zebra crossings in the area). We demonstrate that our cascade classifier is powerful enough to identify most of the zebra crossings, with only a small number of errors that need to be corrected in a subsequent stage of human-supervised image inspection.

5.1. Experimental setting

For the evaluation of the automated pedestrian zebra crossing detection, all experiments were conducted on a laptop computer with *Intel core i7 4500u* 1.8GHz CPU and 8GB RAM. The evaluation considered four spatial regions. The first region, SF1[16], is a dense urban area in San Francisco, composed by wide streets, hit by direct sunlight and surrounded by buildings with 3 to 5 floors. This region was also used in the previous version of this paper [Ahmetovic et al. 2015], and in about 1 year, the layout of the SF1 region changed noticeably as many intersections were repainted with continental crossings (zebra crossings) in place of transverse markings (two line crossings). The number of zebra crossings increased from 141 to 169. However, we noticed that many Street View panoramas were not yet updated to include the new layout, as shown in Figure 9. We constructed a second region, SF2[17], to include and extend the surface covered by SF1 by 200%, and we include both regions to show the impact of the changes in the environment.

The third region, NY[18], located in Manhattan, New York, is characterized by taller buildings which cast shadows on the streets below and limit the visibility of zebra crossings. We selected this region to measure the performance of the automatic detection in low-luminosity conditions. Finally, the fourth region, MI[19], is located in Milan, Italy. This region is composed of a larger number of narrow streets, often occluded by tree branches when viewed in satellite images. This condition has a considerable impact on the recognition and a significant number of zebra crossings are not detected. The web pages in footnotes show the areas with satellite detection results as green polygons. Green pins represent the detected zebra crossings after satellite and Street View detection. False positives (unrelated objects detected as crossings) and false negatives (crossings that are not detected) are shown as yellow and red pins, respectively. Table I reports the main properties of SF1, SF2, NY and MI spatial regions.

Each satellite image has maximum resolution of 640×640 pixels, and given the maximum zoom level available in *Google maps* for the areas considered, this means that each satellite image covers 38m $\times$38m. Thus, a total of 1149 satellite images were required to cover SF1, 3280 to cover SF2, 760 to cover *NY* and 1232 for MI. Since the size of each image is approximately 46KB, the size of all images is 52MB, 151MB, 35MB and 57MB for SF1, SF2, NY, and MI, respectively. The total number of Street View panoramas available in SF1

---

[16]View the area at: http://webmind.di.unimi.it/satzebra/index.html
[17]View the area at: http://webmind.di.unimi.it/satzebra/viewersf.html
[18]View the area at: http://webmind.di.unimi.it/satzebra/viewerny.html
[19]View the area at: http://webmind.di.unimi.it/satzebra/viewermi.html

is 1425, 4294 in SF2, 1606 in NY and 1551 in MI. As we show in the following, we acquired only a small portion of these panoramas, each having a resolution of 640×640 pixels and, on average, a size of 51KB.

The number of zebra crossings reported in Table I was computed using the PCHV web server (see Section 4.4) by one of the authors who thoroughly analyzed satellite and Street View images of each region and manually marked pedestrian zebra crossings, which were defined as specified in Section 4.

In addition to providing the ground truth for the experimental evaluation, we performed a preliminary study of the human supervision validation component of the system for the MI region, as described in Section 5.4. We asked ten participants, recruited among colleagues and friends, to evaluate the labeling produced by our automatic system using the PCHV web service tool (Section 4.4), and to make necessary corrections. We compared the precision and recall values of the labeling before and after human supervision, in order to assess the extent to which intervention by a crowdsourced system can improve the automatically generated labeling.

5.2. Satellite image processing evaluation

As reported in Algorithm 1, only images containing streets are actually considered. With this approach a total of 791 and 2252 images were actually needed for SF1 and SF2 regions, with a total size of 35MB and 103MB respectively, which in both cases is about 67% of total images available. For NY and MI, due to a more dense network of narrower streets, 586 and 1196 images were needed, for a size of 27MB (77%) and 55MB (97%) respectively. This means that, while in high density urban areas most images might be required, in urban contexts where the density of the road network is lower, our technique avoids downloading about one third of the images that would be otherwise required. In suburban or rural areas we can expect that an even higher percentage of potential images would be omitted.

The recognition process described in Algorithm 1 (Lines 7–8) detects a total of 773 zebra crossing portions in SF1, 2866 in SF2, 1243 in NY and 1740 for MI. Often a single zebra crossing is detected as two or more zebra crossing portions. For example this can happen when a vehicle is visible in the middle of the zebra crossing, causing a partial occlusion. By merging these zebra crossing portions (Algorithm 1, Line 12) our technique identifies 199 candidate zebra crossings in SF1, 902 in SF2, 328 in NY and 641 in MI.

The numbers of correct detections (true positives) are reported in Table I. Since we know the number of actual zebra crossings, we can compute the recall of the first step of our classifier, which is 0.972 for SF1, 0.974 for SF2, 0.910 for NY and 0.830 for MI. A few zebra crossings are not detected due to discolored or faded paint (see Figure 10(a)) while others are almost totally covered by trees, shadows or vehicles (see Figure 10(b)). In the NY dataset many zebra crossings are hard to detect due to shadows of buildings severely limiting the luminosity of streets below (see Figure 10(d), zebra crossings outlined in red). The process also yields some wrong detections (false positives), and precision is 0.688, 0.510, 0.646 and 0.565 for SF1, SF2, NY, and MI, respectively. In many cases, false positives correspond to rooftops (Figure 10(c)) or other parts of buildings (Figure 10(e)).

These recall and precision scores refer to parameter settings tuned for high recall so that the least number of zebra crossings is missed by the algorithm. Naturally, perfect recall is difficult to attain and comes at the expense of a greater number of false positives, i.e., a smaller precision. However, considering that a final human supervision validation step is possible, it is much easier for a human supervisor to rule out false positives than it is to find false negatives, which requires scrutinizing the entire area of interest to identify zebra crossings that have not been detected by the algorithm. Parameters can be tuned for different trade-off levels between precision and recall. The Pareto frontier shown in Figure 11 shows the best precision and recall trade-offs obtained during the tuning of the parameters in SF1.

Regarding computation time, we consider the CPU-bound process only and we ignore the time to acquire images, which mainly depends on the quality of the network connection. The CPU-bound computation required for the extraction of candidate zebra crossings in a single image is 180ms. Running the algorithm sequentially on the 791 images acquired for SF1 requires a total of 142s. For SF2, NY and MI the time required is 405s, 105s and 215s, respectively. Note that the process can be easily parallelized and thus it would be straightforward to further reduce computation time.

5.3. Street view image processing evaluation

In SF1 for each candidate zebra crossing in Z (the set of candidate zebra crossings computed with satellite images), there are on average 5.7 nearby Street View panoramas. For SF2 and NY, the results are lower: 4.76 and 4.90 respectively. For MI, in particular the number drops to 2.4 nearby panoramas for each candidate zebra crossing. By considering true positives only (i.e., candidate zebra crossings that represent actual zebra crossings), the average number of nearby Street View panoramas increases to 7.3 (SF1), 5.40 (SF2), 5.20 (NY) and 2.8 (MI). In SF1, only 2 candidate zebra crossings that represent actual zebra crossings have no Street View panoramas in their vicinity (less than 1.5%). Conversely, false positive candidate zebra crossings have a much lower number of nearby Street View panoramas: 2.6 for SF1, 3.3 for SF2, 3.0 for NY and 2.0 for MI. Indeed, in SF1, 19 false positives (30% of the total) do not have any nearby Street View panorama (22% in SF2 and 0% in NY 50% in MI) and 46 false positives, corresponding to 74%, have 3 or fewer surrounding panoramas (52% in SF2, 100% in NY and 100% in MI). This is caused by the fact that, as observed previously, many false positives are located on rooftops or other areas that are not in the immediate vicinity of streets.

As reported in Algorithm 2, our solution acquires nearby panoramas iteratively, until the candidate zebra crossing is validated. In SF1, considering true positives, 1.8 Street View panoramas are acquired on average for each candidate zebra crossing. In 76% of the cases a true positive zebra crossing is validated with at most two panoramas, and up to 56% are validated by processing a single Street View panorama (see Figure 12). Conversely, filtering out false positives requires processing all available nearby Street View panoramas. Overall, the technique requires acquiring and processing ≈ 2 Street View panoramas for each candidate zebra crossing. Similar results can be observed in SF2, NY and MI regions.

The Street View-based validation (tuned for the best recall score) filters out the majority of false positives (see Table II) identified in the previous step, yielding a precision score of

0.971 in SF1, 0.902 in SF2, 0.937 in NY and 0.948 for MI. The few false positives still present are caused by patterns very similar to zebra crossings, like the stairs in Figure 10(e). Considering the true positives, most of them are validated (see Table II again), resulting in a recall score of 0.978 for SF1, 0.928 for SF2, 0.915 for NY and 0.909 for MI. Overall, the recall score of the whole procedure (including both satellite and Street View detection) is 0.950 for SF1, 0.904 for SF2, 0.833 for NY and 0.772 for MI.

As with the satellite detection, different parameter settings yield different precision and recall scores during the validation. Figure 11 shows the settings that yield the best precision and recall trade-offs during the validation.

Regarding computation time, each Street View image can be processed in 46ms and hence the total computation time for is 18.5s, 31.0s, 12.7s and 22s for SF1, SF2, NY and MI, respectively. Overall, considering the two detection steps (from satellite images and Street View panoramas), the total CPU-bound computation time is 161s, 436s, 118s and 238s to process SF1, SF2, NY and MI, respectively.

5.4. Preliminary Study of the Pedestrian Crossing Human Validation

On average, it took $27\,min$ for the participants to scan and update the MI area, with a minimum duration of the experiment of 13 min, 22 sec and a maximum length of 47 min, 7 sec. All participants started with the results of the automated detection results. As shown in Table III, all of the human supervisors managed to drastically reduce the number of errors present after the automated detection steps. In particular, even in the worst case, at least 72% of previously undetected crossings were found, resulting in a recall score of 0.93. This is a great improvement over the previous score of 0.772, obtained with only the automated detection. In the best scenario the resulting recall soars up to 0.997, while the average result is 0.97. Most of the false positive detections have also been pruned through human supervision, resulting in their reduction of at least 40% and an improvement of the precision score from 0.948 to 0.972. In this case the average precision score is 0.989, while Operator 8 managed to prune all false positives and reach a precision score of 1. During the result validation, we noticed that the operators sometimes failed to add crossings or delete false positives in areas that were distant from region borders, intersections, and other markers. We hypothesize that, in absence of points of reference, it is harder to keep track of which areas have already been viewed and processed. In future work we propose the use of smaller validation regions with an automated sliding window mechanism and highlighting of already verified areas.

While most of the modifications by the users were beneficial, in some cases the participants also introduced new errors, like adding non-existent crossings (Figure 13(a)), as in case of Operator 6 or deleting existing ones (Figures 13(b)), such as Operator 3. It is worth noting, however, that the new errors introduced were sporadic and unique among participants. Indeed, every existing zebra crossing was identified by at least one participant, and on average by 7.4 participants, while every false detection was pruned by at least 6 participants, and on average by 7.9. In future work we will perform an extensive evaluation of the human supervision system with a higher number of participants. The goal of the evaluation will be to investigate how to guarantee reliable results for human supervised deletions and insertions

through consensus between human supervisors [Kamar and Horvitz 2012; Kamar et al. 2012].

6. CONCLUSIONS

Blind pedestrians face significant challenges posed by street crossings that they encounter in their everyday travel. While standard O&M techniques allow blind travelers to surmount these challenges, additional information about crosswalks and other important features at intersections would be helpful in many situations, resulting in greater safety and/or comfort during independent travel. We investigated the role that crosswalks play in travel by blind pedestrians, and the value that additional information about them could add, in a survey conducted with nine O&M experts. The results of the survey show stark differences between survey respondents from the U.S. compared with Italy: the former group emphasized the importance of following standard O&M techniques at all legal crossings (marked or unmarked), while the latter group strongly recommended crossing at marked crossings whenever possible. These contrasting opinions reflect differences in the traffic regulations of the two countries and highlight the diversity of needs that travelers in different regions may have.

To address the challenges faced by blind pedestrians in negotiating street crossings, we devised a computer vision-based technique that mines existing spatial image databases for discovery of zebra crosswalks in urban settings. Our algorithm first searches for zebra crosswalks in satellite images; all candidates thus found are validated against spatially registered Google Street View images. This cascaded approach enables fast and reliable discovery and localization of zebra crosswalks in large image datasets. We evaluated our algorithm on four urban regions in which the pedestrian crossings were manually labeled, including two in San Francisco, one in New York City and the fourth in Milan, Italy, covering a total area of $7.5 km^2$. The technique achieved a precision ranging from 0.902 to 0.971, and a recall ranging from 0.772 to 0.950, over the four regions.

While fully automatic, our algorithm can be improved by a final crowdsourcing validation. To this end, we developed a Pedestrian Crossing Human Validation (PCHV) web service, which supports human validation to rule out false positives and identify false negatives.

Our approach combines computer vision and human validation to detect and localize zebra crosswalks with high accuracy in existing spatial images databases such as Google satellite and Street View. To the best of our knowledge, these features are not already marked in existing GIS services and we argue that knowing the position of existing crosswalks could be useful to travelers with visual impairments for navigation planning, finding pedestrian crosswalks and crossing roads.

In the future, we plan to extend our approach to detect other common types of crosswalk markings, such as transverse lines. The crossings database could also be augmented with auxiliary information such as the presence and location of important features such as walk lights (which could be monitored in real time by the app) and walk push buttons. These could be also added by users or automatically by a visual detection app.

The resulting crosswalk database could be accessed by visually impaired pedestrians through the use of a GPS navigation smartphone app. The app could identify the user's current coordinates and look up information about nearby zebras, such as their number, placement and orientation. Existing apps such as Intersection Explorer and Nearby Explorer (for Android) and The Seeing Eye GPS[TM] and ClickAndGo (for iPhone) could be modified to incorporate information from the crosswalk database. Computer vision-based detection apps on smartphones, such as Zebralocalizer [Ahmetovic et al. 2014] and Crosswatch [Coughlan and Shen 2013], could additionally use the database to help users to approach and align properly to crosswalks even when they are not yet detected by the app.

Another important way in which the crosswalk database could be used by blind and visually impaired pedestrians is for help with offline route planning, which could be conducted by a traveler on his/her smartphone or computer from home, work or other indoor location before embarking on a trip. For example, a route-planning algorithm may weigh several criteria in determining an optimal route, including the number of non-zebra crossings encountered on the route (which are less desirable to traverse than zebra crossings) as well as standard criteria such as total distance traversed. The crosswalk database could also be augmented with additional useful information, including temporary hazards or barriers due to road construction, etc.

Acknowledgments

References

Ahmetovic, Dragan, Bernareggi, Cristian, Gerino, Andrea, Mascetti, Sergio. Int Conf on Pattern Recognition. IEEE; 2014. ZebraRecognizer: Efficient and Precise Localization of Pedestrian Crossings.

Ahmetovic, Dragan, Bernareggi, Cristian, Mascetti, Sergio. Zebralocalizer: identification and localization of pedestrian crossings. Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services; ACM; 2011. p. 275-284.

Ahmetovic, Dragan, Gleason, Cole, Kitani, Kris, Takagi, Hironobu, Asakawa, Chieko. NavCog: turn-by-turn smartphone navigation assistant for people with visual impairments or blindness. Web for All Conference; ACM; 2016a.

Ahmetovic, Dragan, Gleason, Cole, Ruan, Chengxiong, Kitani, Kris, Takagi, Hironobu, Asakawa, Chieko. NavCog: A Navigational Cognitive Assistant for the Blind. International Conference on Human Computer Interaction with Mobile Devices and Services; ACM; 2016b.

Ahmetovic, Dragan, Manduchi, Roberto, Coughlan, James M., Mascetti, Sergio. Zebra Crossing Spotter: Automatic Population of Spatial Databases for Increased Safety of Blind Travelers. Proceedings of the 17th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2015); 2015.

Akinlar C, Topal C. EDLines: A real-time line segment detector with a false detection control. Pattern Recognition Letters. 2011

Ancillotti, Massimo, Carmagnini, Giuseppe. La riforma del codice della strada. La nuova disciplina dopo la legge 120/2010. Maggioli Editore. 2010

Ashmead, Daniel H., Guth, David, Wall, Robert S., Long, Richard G., Ponchillia, Paul E. Street crossing by sighted and blind pedestrians at a modern roundabout. Journal of Transportation Engineering. 2005; 131(11):812–821.

Barlow JM, Bentzen BL, Sauerburger D, Franck L. Teaching travel at complex intersections. Foundations of Orientation and Mobility. 2010

Campbell, Megan, Bennett, Cynthia, Bonnar, Caitlin, Borning, Alan. Where's my bus stop?: supporting independence of blind transit riders with StopInfo. Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility; ACM; 2014. p. 11-18.

Coughlan, James M., Shen, H. Crosswatch: a system for providing guidance to visually impaired travelers at traffic intersection. Jour of Assistive Technologies. 2013

Fitzpatrick, Kay, Chrysler, Susan T., Iragavarapu, Vichika, Park, Eun Sug. Crosswalk marking field visibility study. Technical Report. 2010

Fusco, Giovanni, Shen, Huiying, Coughlan, James M. Self-Localization at Street Intersections. Computer and Robot Vision (CRV), 2014 Canadian Conference on; IEEE; 2014a. p. 40-47.

Fusco, Giovanni, Shen, Huiying, Murali, Vidya, Coughlan, James M. Determining a blind pedestrians location and orientation at traffic intersections. International Conference on Computers for Handicapped Persons; Springer; 2014b. p. 427-432.

Goodchild, Michael F. Citizens as sensors: the world of volunteered geography. GeoJournal. 2007

Guy, Richard, Truong, Khai. Conf on Human Factors in Computing Systems. ACM; 2012. CrossingGuard: exploring information content in navigation aids for visually impaired pedestrians.

Hara, Kotaro, Azenkot, Shiri, Campbell, Megan, Bennett, Cynthia L., Le, Vicki, Pannella, Sean, Moore, Robert, Minckler, Kelly, Ng, Rochelle H., Froehlich, Jon E. Int Conf on Computers and Accessibility. ACM; 2013a. Improving public transit accessibility for blind riders by crowdsourcing bus stop landmark locations with Google street view.

Hara, Kotaro, Le, Vicki, Froehlich, Jon. Conf on Human Factors in Computing Systems. ACM; 2013b. Combining crowdsourcing and google street view to identify street-level accessibility problems.

Hara, Kotaro, Le, Victoria, Sun, Jin, Jacobs, David, Froehlich, J. Exploring Early Solutions for Automatically Identifying Inaccessible Sidewalks in the Physical World Using Google Street View. HCI Consortium. 2013c

Hara, Kotaro, Sun, Jin, Moore, Robert, Jacobs, David, Froehlich, Jon. Tohme: detecting curb ramps in google street view using crowdsourcing, computer vision, and machine learning. Proceedings of the 27th annual ACM symposium on User interface software and technology; ACM; 2014. p. 189-204.

Hochmair, Hartwig H., Zielstra, Dennis, Neis, Pascal. Assessing the completeness of bicycle trails and designated lane features in OpenStreetMap for the United States and Europe. Transportation Research Board Annual Meeting; 2013.

Kacorri, Hernisa, Mascetti, Sergio, Gerino, Andrea, Ahmetovic, Dragan, Takagi, Hironobu, Asakawa, Chieko. Supporting Orientation of People with Visual Impairment: Analysis of Large Scale Usage Data. Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility; ACM; 2016. p. 151-159.

Kamar, Ece, Hacker, Severin, Horvitz, Eric. Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1. International Foundation for Autonomous Agents and Multiagent Systems; 2012. Combining human and machine intelligence in large-scale crowdsourcing; p. 467-474.

Kamar, Ece, Horvitz, Eric. Proceedings of the 11th international conference on autonomous agents and multiagent systems-volume 3. International Foundation for Autonomous Agents and Multiagent Systems; 2012. Incentives for truthful reporting in crowdsourcing; p. 1329-1330.

Kirschbaum, Julie B., Axelson, Peter W., Longmuir, Patricia E., Mispagel, Kathleen M., Stein, Julie A., Yamada, Denise A. Designing Sidewalks and Trails for Access, Part 2, Best Practices Design Guide. Technical Report. 2001

Knoblauch, Richard L., Crigler, Kristy L. Model pedestrian safety program user's guide. Technical Report. 1987

Koester, Daniel, Lunt, Björn, Stiefelhagen, Rainer. Zebra Crossing Detection from Aerial Imagery Across Countries. Proc. of the 15th Int. Conf. on Computer Helping People with Special Needs; IEEE; 2016. To appear

Kubásek, Miroslav, H ebí ek, Ji í., et al. Crowdsource Approach for Mapping of Illegal Dumps in the Czech Republic. Int Jour of Spatial Data Infrastructures Research. 2013

Leitloff, Jens, Hinz, Stefan, Stilla, Uwe. Vehicle detection in very high resolution satellite images of city areas. Trans on Geoscience and Remote Sensing. 2010

Lin, Tsung-Yi, Cui, Yin, Belongie, Serge, Hays, James. Learning deep representations for ground-to-aerial geolocalization. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); IEEE; 2015. p. 5007-5015.

Mascetti, Sergio, Ahmetovic, Dragan, Gerino, Andrea, Bernareggi, Cristian. ZebraRecognizer: Pedestrian Crossing Recognition for People with Visual Impairment or Blindness. Pattern Recognition. 2016

Mattyus, Gellert, Wang, Shenlong, Fidler, Sanja, Urtasun, Raquel. Enhancing Road Maps by Parsing Aerial Images Around the World. Proceedings of the IEEE International Conference on Computer Vision; 2015. p. 1689-1697.

Mokhtarzade, Mehdi, Valadan Zoej, MJ. Road detection from high-resolution satellite images using artificial neural networks. Int jour of applied earth observation and geoinformation. 2007

Murali, V., Coughlan, James M. Int Conf on Multimedia and Expo (workshop). IEEE; 2013. Smartphone-Based Crosswalk Detection and Localization for Visually Impaired Pedestrians.

Poggi, Matteo, Nanni, Luca, Mattoccia, Stefano. Crosswalk Recognition Through Point-Cloud Processing and Deep-Learning Suited to a Wearable Mobility Aid for the Visually Impaired. New Trends in Image Analysis and Processing–ICIAP 2015 Workshops; Springer; 2015. p. 282-289.

Pogrund, Rona L., et al. Texas School for the Blind and Visually Impaired. 2012. Teaching age-appropriate purposeful skills: An orientation & mobility curriculum for students with visual impairments.

Rice, Matthew T., Aburizaiza, Ahmad O., Daniel Jacobson, R., Shore, Brandon M., Paez, Fabiana I. Supporting Accessibility for Blind and Vision-impaired People With a Localized Gazetteer and Open Source Geotechnology. Transactions in GIS. 2012

Rituerto, Alejandro, Fusco, Giovanni, Coughlan, James M. Towards a Sign-Based Indoor Navigation System for People with Visual Impairments. Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility; ACM; 2016. p. 287-288.

Senlet, Turgay, Elgammal, Ahmed. Int Conf on Pattern Recognition. IEEE; 2012. Segmentation of occluded sidewalks in satellite images.

Sirmacek, Beril, Unsalan, Cem. Urban-area and building detection using SIFT keypoints and graph theory. Trans on Geoscience and Remote Sensing (2009). 2009

Wiener, William R., Welsh, Richard L., Blasch, Bruce B. Foundations of orientation and mobility. American Foundation for the Blind; 2010.

Xiao, Jianxiong, Quan, Long. Int Conf on Computer Vision. IEEE; 2009. Multiple view semantic segmentation for street view images.

Zamir, Amir Roshan, Shah, Mubarak. Oroc of European Conf on Computer Vision. Springer; 2010. Accurate image localization based on google maps street view.

Zegeer, Charles V. Design and safety of pedestrian facilities. Institute of Transportation Engineers; 1995.

Appendix: Survey Questions

Question 1 (U.S. version)

As you know, while many crosswalks are marked by painted lines, others crosswalks at street intersections are not. According to the Federal Highway Administration, marked crosswalks are used to "define the pedestrian path of travel across the roadway and alert

drivers to the crosswalk location". *Do you think, based on your experience, that blind travelers should preferentially cross a street on marked crosswalks?*

Question 1 (Italian version)

As you know, in Italy pedestrians must cross a street on a marked crosswalk, provided that one is available within 100m from the crossing position. Regardless of the crossing location, vehicles should always yield to blind pedestrians with a white cane or a guide dog. *Do you think, based on your experience, that blind travelers should preferentially cross a street on marked crosswalks?*

Question 2

Suppose that you are helping an independent blind traveler plan a path that requires crossing one or more streets. Consider the following hypothetical case. The shortest path (path 1) requires crossing a street at an intersection where there are no marked crosswalks (but where crossing is legal). However, it would be possible to cross the same street on a marked crosswalk that is placed mid-block; this would result in a longer path (path 2). (See Fig. 14.) *Which one of the two paths would you recommend?* In your answer, please consider two different cases. In the first case, the traveler, during the trip, will need to figure out by himself where the crosswalk is. In the second case, the traveler has an app in his smartphone that helps him find out the location of the marked crosswalk. Also, please evaluate which factors would determine whether the traveler should prefer one path over the other.

Question 3

Suppose that a person who is blind is going to a destination she has never visited before. This blind traveler has not planned a complete path beforehand, or, due to unexpected circumstances, had to take a detour. At some point, she realizes that she needs to cross a street she is not familiar with. *What type of information would she need to access in order to determine where to cross the street, that she would be unable to obtain by herself? In your opinion, are existing apps (e.g. Blindsquare) capable of providing this type of information? What other type of information that is not currently achievable by an existing app would be useful to this blind traveler?*

Question 4

Suppose you were to advise the designer of an app that gives a person who is blind information about a specific street intersection. *What would be the most relevant information that this app should be able to provide?* In this answer, please consider the unique challenges posed to blind travelers by complex intersections, or intersections with non-standard and/or unexpected features.
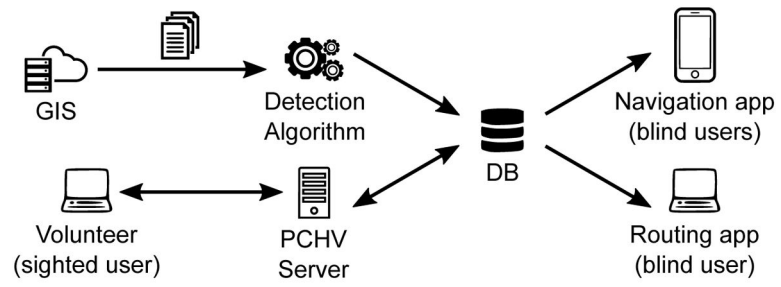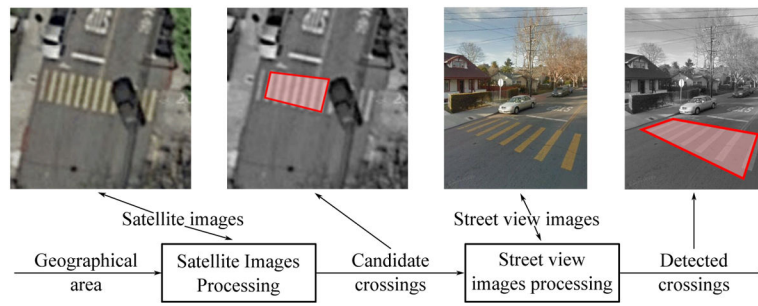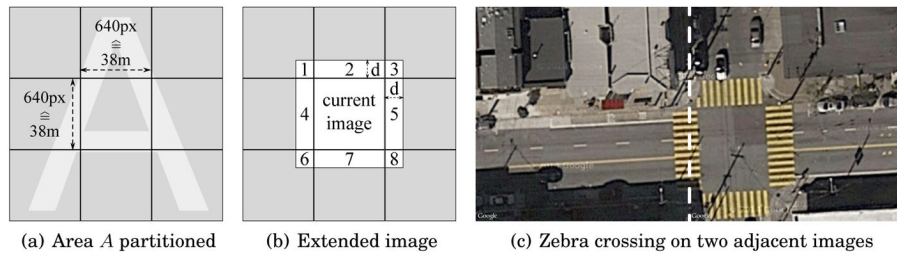
Fig. 1.
System architecture.

Fig. 2.
Overview of computer vision cascade approach: the system finds candidate crosswalk markings in satellite images and then verifies or rejects these candidates in the corresponding Street View images.

(a) Area A partitioned   (b) Extended image   (c) Zebra crossing on two adjacent images

Fig. 3.
The system collects satellite images of the target area.

(a) Original image   (b) Line segments   (c) Stripes   (d) Crossing
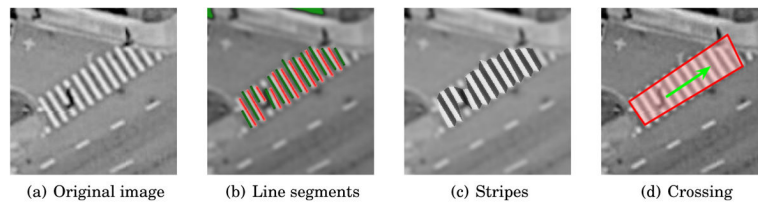
Fig. 4.
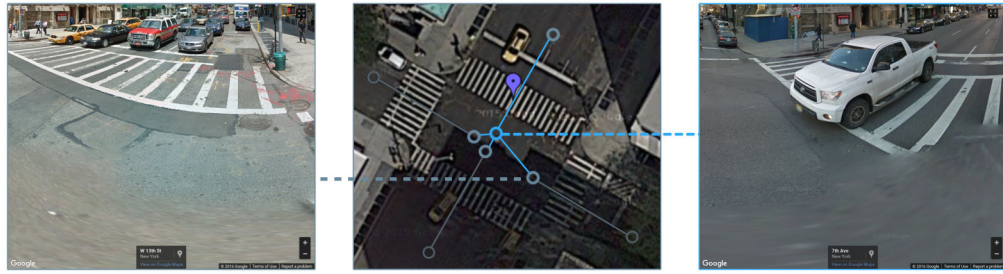The system recognizes a zebra crossing from satellite images in four main steps.

Fig. 5.
In the center, the satellite image with the purple pin marking the detected crossing and the graph of surrounding street views. On the right, the closest Street View shows the crossing covered by a car. On the left, the crossing is visible.
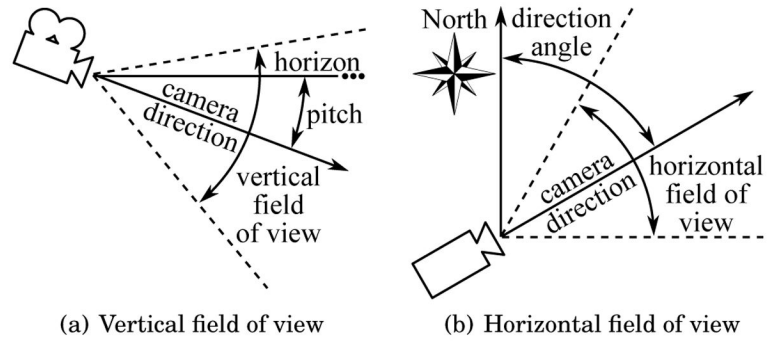
(a) Vertical field of view (b) Horizontal field of view

Fig. 6.
Horizontal and vertical field of view

(a) Satellite detection (b) Streetview detection (c) Reconstructed aerial view (d) Detection imprecision
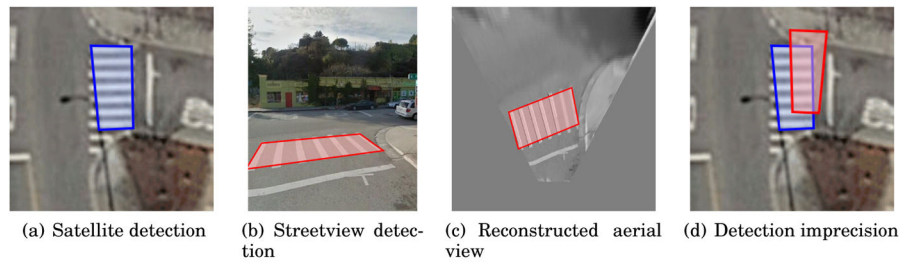
Fig. 7.
Imprecision in GPS coordinates between satellite and Street View detected crossings.
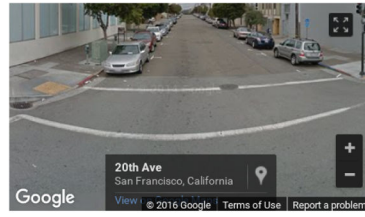
Fig. 8.
A screenshot of the PCHV web service. On the left side, the satellite image pane and, on the right side, Street View pane.

(a) Satellite image shows a zebra crossing (b) Street view still shows transverse markings

Fig. 9.
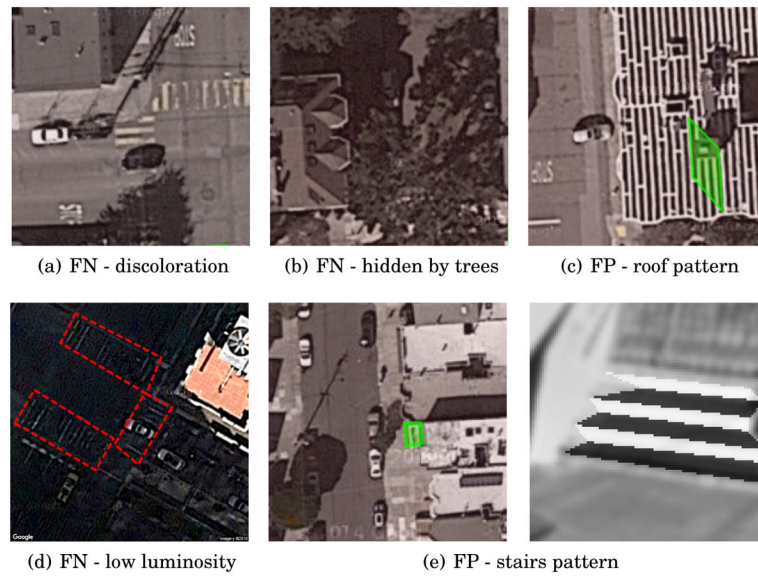Inconsistency between satellite and Street View data.

(a) FN - discoloration   (b) FN - hidden by trees   (c) FP - roof pattern

(d) FN - low luminosity   (e) FP - stairs pattern

Fig. 10.
False negatives (FN) and false positives (FP) in satellite detection and Street View validation.

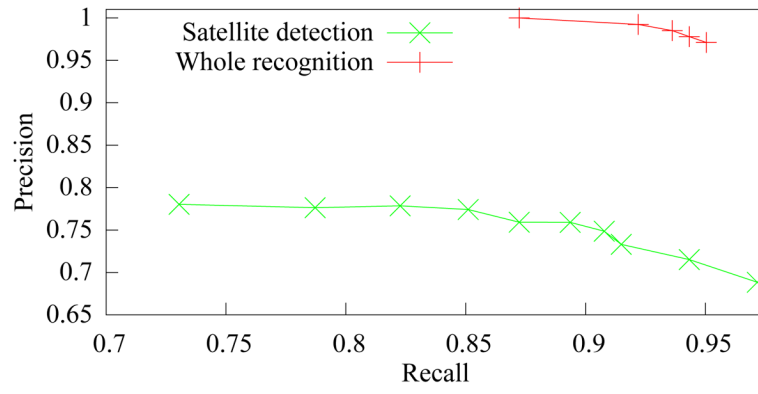**Fig. 11.**
Pareto frontier of the detection procedure for SF1
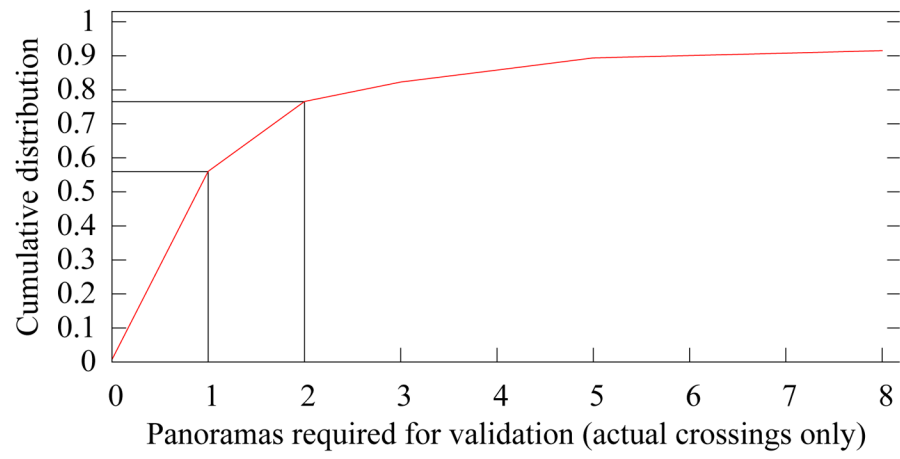
Fig. 12.
Used panoramas cumulative distribution for SF1.

(a) Operator 6: Insertion Errors (b) Operator 3: Deletion Errors

Fig. 13.
False positives (FP) and false negatives (FN) in satellite detection and Street View validation.
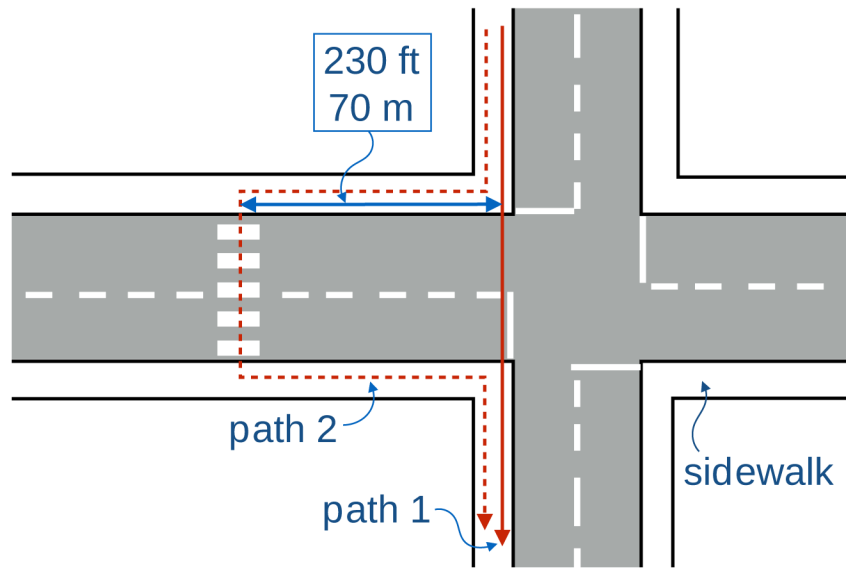
Fig. 14.
An hypothetical street crossing scenario.

Table I

Main properties of the three spatial regions used in the experimental evaluation.

Region	Area	Zebra crossings	Satellite Images		Street View Panoramas	
			Available	Downloaded	Available	Downloaded
SF1	1.66km^2	141	1149	791	1425	964
SF2	4.74km^2	461	3280	2252	4924	867
NY	1.10km^2	254	760	586	1606	379
MI	1.69km^2	436	1232	1196	1551	483

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

Table II

Evaluation results for the three regions. The metrics considered are TP: true positives, FP: false positives, FN: false negatives, Precision and Recall.

Region	Satellite Detection					Street View Validation				
	TP	FN	FP	Precision	Recall	TP	FN	FP	Precision	Recall
SF1	137	4	62	0.688	0.972	134	7	4	0.971	0.950
SF2	458	12	444	0.510	0.974	425	45	46	0.902	0.904
NY	212	21	116	0.646	0.910	194	39	13	0.937	0.833
MI	362	74	279	0.565	0.830	329	107	20	0.948	0.772

Table III

Results of the preliminary evaluation of the Pedestrian Crossing Human Validation component for the MI region. The metrics considered are the true positive (TP), false positive (FP), false negative (FN) crossings, and Precision and Recall scores after joint automated detection and human validation. Insertion errors and Deletion errors define FP and FN crossings, respectively, that were caused by participants' errors.

Operator	TP	FP	FN	Precision	Recall	Insertion Errors	Deletions Errors	Duration
1	432	4	2	0.995	0.991	1	0	20 min
2	427	9	8	0.982	0.979	2	0	34 min
3	406	30	1	0.997	0.931	0	4	13 min
4	420	16	12	0.972	0.963	1	0	26 min
5	428	8	2	0.995	0.982	0	0	19 min
6	428	8	8	0.982	0.982	4	0	47 min
7	415	21	8	0.981	0.952	0	0	30 min
8	428	8	0	1	0.982	0	0	34 min
9	424	12	5	0.988	0.972	0	1	20 min
10	433	3	3	0.995	0.993	0	0	55 min