

Article

Real-Time Sidewalk Slope Calculation through Integration of GPS Trajectory and Image Data to Assist People with Disabilities in Navigation

Yihan Lu * and Hassan. A. Karimi

School of Information Sciences, University of Pittsburgh, 135 North Bellefield Avenue, Pittsburgh, PA 15260, USA; E-Mail: hkarimi@sis.pitt.edu

* Author to whom correspondence should be addressed; E-Mail: lyihan@asu.edu;
Tel.: +1-412-888-9982.

This is an extended version of presentation on TIC'14: The ISPRS Tracking and Imaging Challenge 2014.

Academic Editors: Stephan Winter, Alper Yilmaz, Monika Sester and Wolfgang Kainz

Received: 31 January 2015 / Accepted: 27 April 2015 / Published: 4 May 2015

Abstract: People with disabilities face many obstacles in everyday outdoor travels. One of the most notable obstacles is steep slope on sidewalk segments. Current navigation systems/services do not all support map databases with slope attributes and cannot calculate sidewalk slope in real time. In this paper, we present a technique for calculating slopes of sidewalk segments by image data and predict the most suitable route for each individual user through integration with GPS trajectory. In our technique we make use of GPS trajectory data, to identify the sidewalk segment on which the traveler will most probably pass, and images of the identified sidewalk segment. Through edge detection techniques we detect edges of objects, such as buildings, billboards, and walls, in the background. Slope of the segment is then calculated by comparing its line representation in the map with the detected edges. Our experiment result indicates effective calculation of sidewalk slopes.

Keywords: GPS trajectory; image processing; navigation

1. Introduction

Independent mobility is an important aspect of quality of life for people with disabilities (PWDs). In particular, people who use wheelchairs and are blind or visually impaired frequently encounter unique challenges when traveling outdoors [1]. Navigation systems/services could play an important role in assisting PWDs with their mobility needs and references. Meyers *et al.* interviewed some wheelchair users for the types of obstacles they usually encounter when traveling outdoors and found that narrow sidewalks, no ramps, steep ramps, no sidewalk curb cuts, poor sidewalk surfaces, and sidewalk obstructions, are among obstacles [2]. A navigation system/service designed to address the mobility challenges of PWDs would include information about the accessibility of the sidewalks in its database and assist in avoiding obstacles. In this paper, we present a technique to calculate sidewalk slope, one type of obstacle that when it is steep would impede travel by PWDs, in real time. In our technique, the user trajectory is used to access nearby sidewalk images from the Google Street View image data set, and the slope of the sidewalk is calculated utilizing edges, detected by our technique, of buildings in the background. An important feature of our technique is that slope is calculated using only a single image.

Our proposed technique is novel and its major contribution is providing a service for PWDs to avoid steep slopes through calculating sidewalk slope by image data, analyzing the trajectory to determine user's preference, and recommending a suitable route close to the user. The technique can be incorporated into existing or new navigation systems/services. We demonstrate the feasibility and accuracy of this technique through an experiment. The structure of the paper is as follows. Related works are discussed in Section 2. The proposed technique, integrating GPS trajectory data and image data, to calculate sidewalk slope in real time is described in Section 3. A recommendation method considering both the slope and the route length is proposed in Section 4. In Section 5, the result of an experiment using the technique is discussed. Conclusion and future research are discussed in Section 6.

2. Relate Works

Smith *et al.* proposed a novel framework that processes, identifies, and classifies sidewalks in Google Street View images [3]. They use image processing techniques to determine the horizontal and vertical components, and consider the color, location, and presence of lanes as the cues to detect sidewalks. Senlet *et al.* developed a framework to construct sidewalk and crosswalk maps from satellite images [4]. These works differ from the work in this paper in that they only detect sidewalks in images, while our contribution is calculating slope of a sidewalk after detecting it in images.

Alexandra *et al.* developed the Automated Sidewalk Quality Assessment Tool, a low-cost system that can be used to automatically assess sidewalk quality and can be deployed by agencies on a large scale [5]. Their system requires installation of App-equipped tablets on wheelchairs, so that when a wheelchair is roving, the tablet records video, GPS location, accelerometer, and gyroscope data. Then the video and vibration data are analyzed for areas requiring repair. Through image processing techniques, they calculated the parameters of sidewalks including width, pavement type, surface quality (e.g., roughness, discontinuities, and gaps), grade, and number of obstructions. They also developed a complete sidewalk quality index for the city of Atlanta. Their work differs from ours in

that they utilized specific devices to collect composite data about sidewalks, while we only make use of sidewalk network and Google Street View image data to calculate sidewalk slope in real time.

Ren and Karimi developed a navigation application on Android to assist wheelchair users with navigating on sidewalks. They employed GPS, accelerometer, compass, and camera on Android devices to find real-time position of wheelchair users [6]. Their work is different than ours in that they do not calculate sidewalk slope in real time.

In summary, while in some existing works sidewalks in images are detected and camera images are used to map match user's position to sidewalk segments, there is a void in literature on calculating sidewalk slopes in real time. In this paper, we address this gap by developing a sidewalk slope calculation technique.

3. Slope Calculation Technique

The proposed technique has four steps: identifying user's position on the sidewalk segment using GPS trajectory, finding images from Google Street View that include the sidewalk segment, detecting the sidewalk segment in each image, and calculating the slope of the sidewalk. The steps and associated algorithms are illustrated in Figure 1.

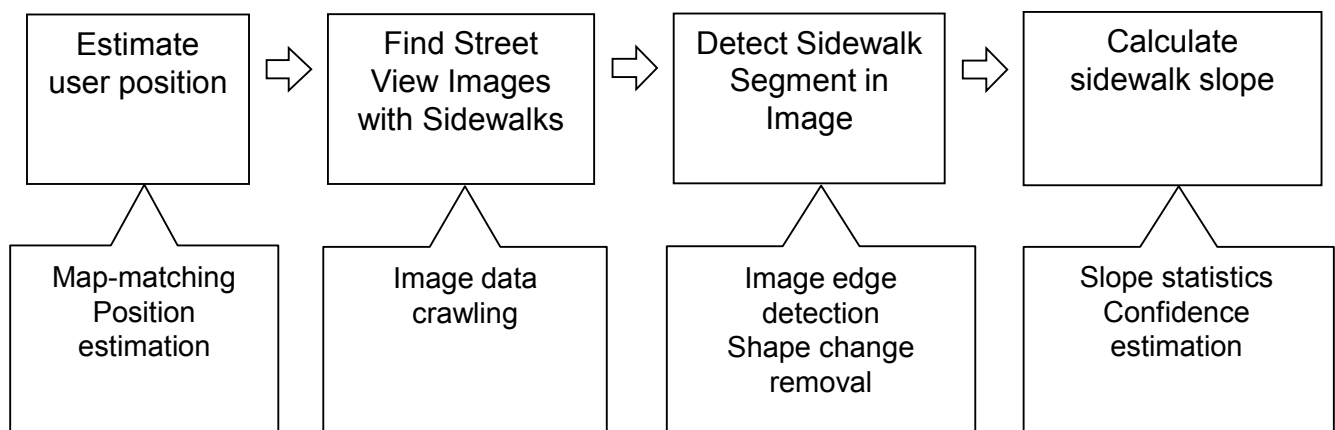


Figure 1. The steps and associated algorithms for the slope calculation technique.

3.1. Estimating User's Position on Sidewalk

The first step of our technique map matches user's position on the sidewalk segment using GPS data, *i.e.*, GPS map matching [7,8]. Map-matching algorithms have extensively been investigated, which is one of the critical techniques in personal navigation assistants [7]. In our technique we use the algorithm developed by Ren and Karimi since it is designed for sidewalk segment matching [6]. This algorithm is based on Hidden Markov Model and is chosen over the other potential algorithms, such as fuzzy-logic and chain-coding, because it can produce high accuracy results [6]. Highly accurate map-matching algorithms, such as the one we have chosen for our technique, provides a better estimate of user's position on a sidewalk network and makes it possible to identify nearby sidewalk segments, both needed for real-time guidance of PWDs. After accurately matching the user's GPS point onto the sidewalk map, a buffer is generated around the user's position to find the next sidewalk segment en route. This is done by taking into account the user's previous position, current velocity direction, and

accuracy of the user's current position [6]. We use the Personalized Accessibility Map (PAM) database, which includes the sidewalk network of the University of Pittsburgh's main campus, to experiment with our technique. Figure 2 shows the sidewalk network in PAM [6,9].

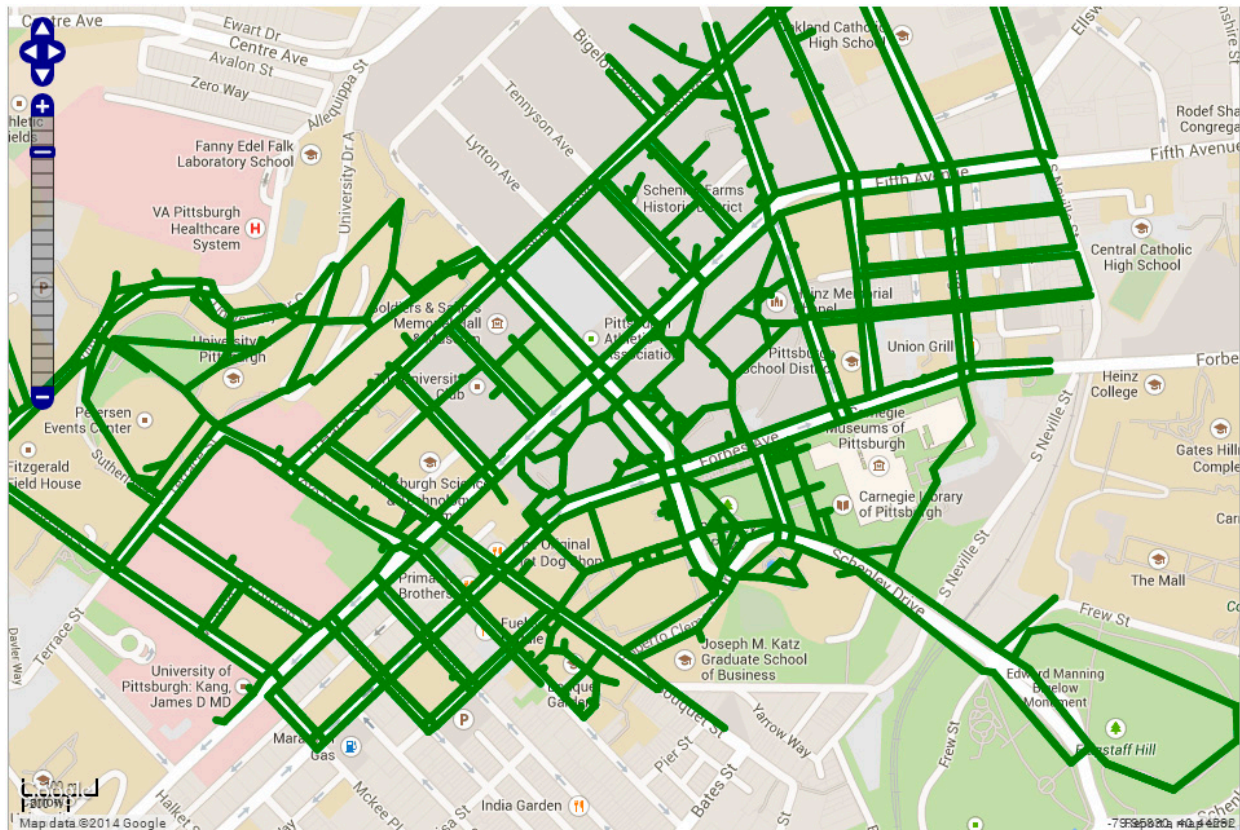


Figure 2. Sidewalk network in Personalized Accessibility Map (PAM).

3.2. Finding Google Street View Images with Sidewalks

In the second step, by providing a position's coordinates (latitude, longitude) and sight direction, the technique searches for images from Google Street View and selects those that include the identified sidewalk segment in the first step. We used Google Street View as the source of images in this work, and while they are not of very high quality, they are freely available for many geographic areas. An alternative source of images is on-board cameras (e.g., cameras on smartphones), but despite their high quality, capturing these images poses practical concerns. Example concerns for calculating the slope of one segment are: How many images should be captured? Where the images should be captured? At what angle and distance each image should be captured?

Figure 3 shows an example image describing a part of the sidewalk segment. It is important for the technique that the sight direction of the camera, when taking the image, is perpendicular to the sidewalk segment. The parallax shape change is minimized in this sight direction, and it is ignored if the viewed object is at the center of the image.

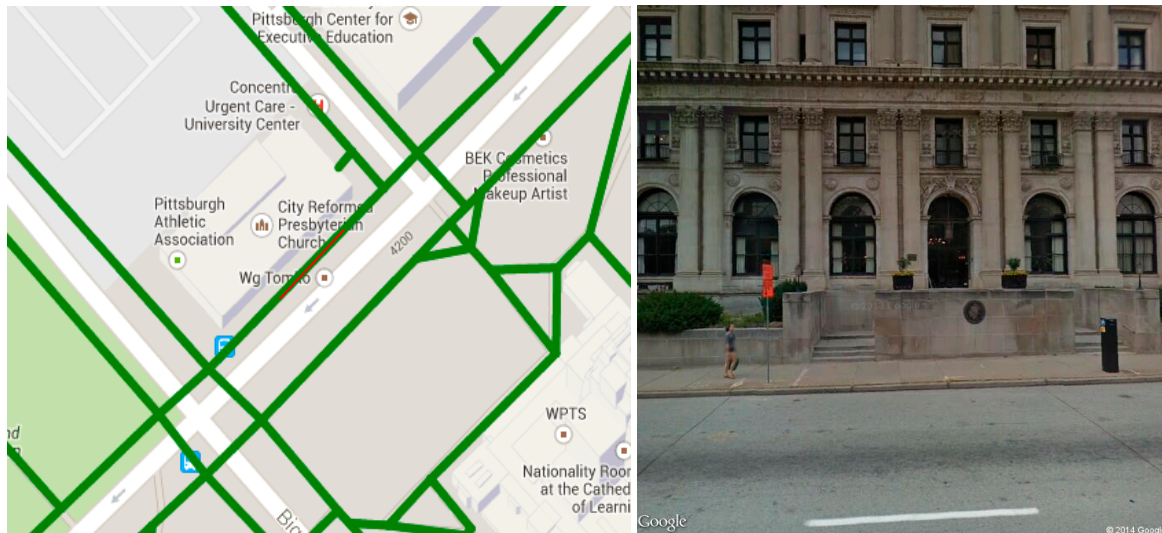


Figure 3. Example Google Street View image containing a sidewalk segment.

3.3. Detecting Sidewalk Segment in Image

In the third step of the technique, the identified sidewalk segment's slope is detected. Smith *et al.* made use of color, location, and presence of lanes in Google Street View images to detect sidewalks, while the sight of the Google Street View image aligns the street [3]. Senlet *et al.* used satellite images to detect sidewalk networks [4]. However, neither of these techniques can be reused in our work because slope cannot be detected using their sight direction. Figure 4 shows sidewalks detected by these other techniques. In our technique, we set the sight direction perpendicular to the sidewalk to reduce the parallax shape change. More importantly, perpendicular sight direction can help detect the slope of sidewalk by comparing it against the buildings in the background.



Figure 4. Sidewalk detection techniques: (a) sidewalks detected (blue) in Google Street View; image (b) segments (red) detected as sidewalks.

Since our technique calculates only sidewalk slope which is the same as the slope of the road connected to it, it is not necessary to partition the sidewalk network. As a result, any straight line detected along the road can be used to calculate the slope. In our technique, we use Canny's algorithm to detect edges in Google Street View images, and then use Hough transfer to detect straight lines in the edges [10–12]. The lowest long horizontal straight line is considered to be the sidewalk. Figure 5 shows the result of the sidewalk detection step.

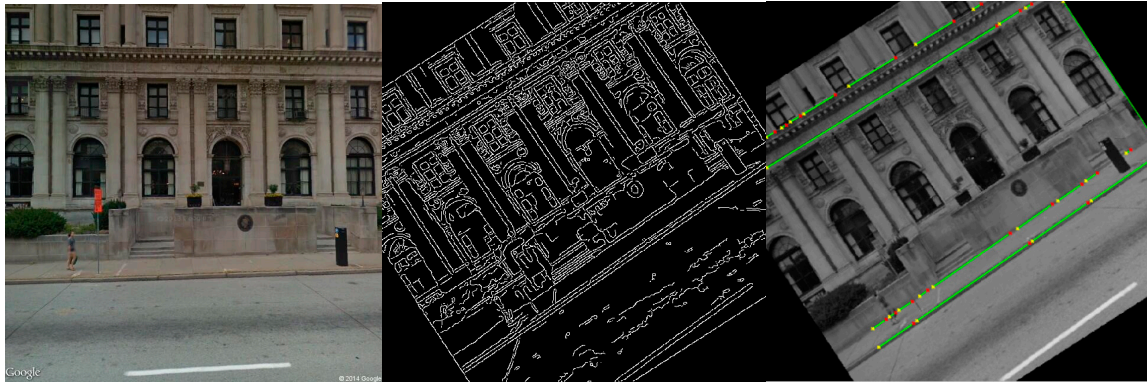


Figure 5. Sidewalk detection. The lowest line on the right image is used to calculate the angle of the sidewalk segment.

3.4. Calculating Slope

Detected straight lines in Google Street View images include sidewalks and other straight lines in the background. The trend among the straight lines indicates the degree of slope of the current sidewalk. Figure 6 shows two examples of straight lines detected on the buildings in the background. The straight lines on buildings are usually horizontal and vertical and are used in our technique as a reference to calculate sidewalks slopes.

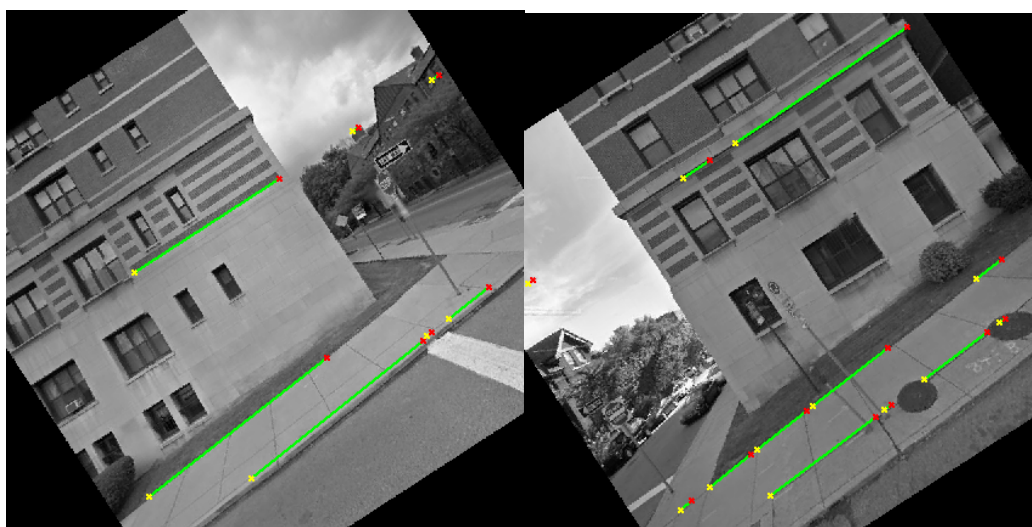


Figure 6. Examples of straight line detection. The slope of the sidewalk segment is the angle between the line representing the sidewalk segment and the lines representing the building.

Assuming the parallax shape change can be ignored, the sidewalk is detected correctly and the straight line in the background is horizontal; the slope of the sidewalk is the angle between the sidewalk (straight line) and the background straight line. Let P_1 and P_2 denote two points on the edge of a sidewalk segment and Q_1 and Q_2 denote two points on the edge of the background building. Then the angle of the slope can be calculated as follows:

$$slope = \arccos \left\{ \frac{\overline{P_1P_2} \cdot \overline{Q_1Q_2}}{|\overline{P_1P_2}| \times |\overline{Q_1Q_2}|} \right\} \tag{1}$$

4. Slope Network Profile

Once the slope of a segment is calculated in real time, it is recorded as part of the user’s GPS trajectory, formed by a sequence of adjacent sidewalk segments. As the user continues traveling on new segments, the slopes on the new segments are calculated and a slope network profile for each trajectory can be constructed. The constructed slope network profile can assist in tracking user’s position and predicting best possible routes for the user during real-time navigation. Figure 7 shows an example of a user’s trajectory along with three slope network profiles, each for a different segment of a route. These examples are generated manually to show how slope network profiles can be used in predicting routes.



Figure 7. (a) Example trajectory history for a user; (b) An elevation trend curve indicating user’s pattern of slope usage.

Trajectory history of a user includes slope network profiles that indicate the user’s slope pattern of usage and preference. Such patterns of usage and preference of slope can be used in finding and recommending personalized routes in real time. It is worth noting that for evaluating trajectory slopes and finding and recommending personalized routes, it is important to pay attention to slope trends in routes and identify those with relatively uniform trends. Figure 8 shows two routes with the same origin and destination. Although both origin and destination locations are at the same elevation, the

slope of the route in blue is relatively uniform, while the route in red has more steep segments. In this example, the route in blue is more suitable for PWDs compared to the route in red.

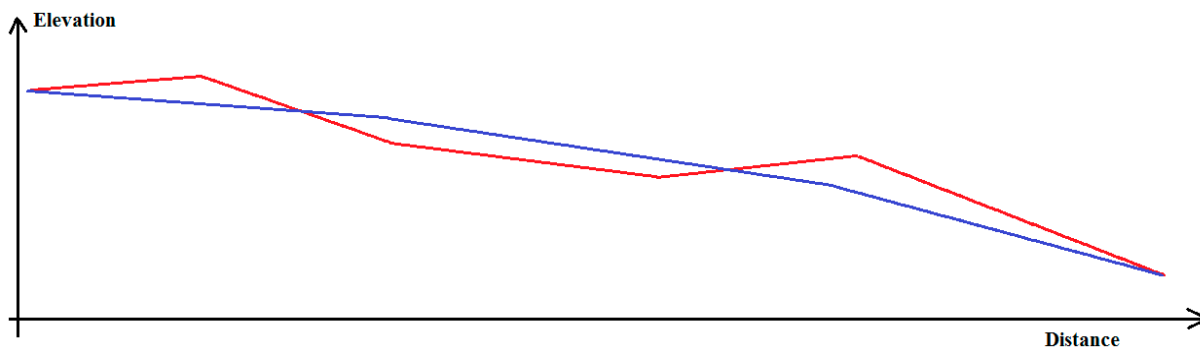


Figure 8. Two example routes with the same pair of origin and destination.

The proposed approach above requires development of a method to calculate a value to represent the slope trend of routes, while avoiding the bias caused by the total length of the route and the general elevation of the route. Based on this observation, the variance of the slope can correctly represent the trend of a trajectory, which can be expressed as follows:

$$Var = \sigma^2 = \int x^2 E_x dx - \mu^2 \tag{2}$$

where x is the distance from the location of the origin and E_x is the elevation at position x . μ is the average slope, calculated as follows:

$$\mu = (E_d - E_o) / l \tag{3}$$

where E_d and E_o are the elevations at the destination and origin, respectively, and l is the total length of the route.

In Equation (1), the lower Var is, the smoother the slope of the route will be. After calculating the Var values for multiple trajectories of the same user, the slope that the user prefers can be estimated which in turn can be used in updating routes.

By integrating the user’s preferred slope with other route attributes, a method to recommend personalized routes can be developed. Of various route attributes, route length is of particular importance for developing such a method. In our method, we integrate length, slope variance, and user’s preference to find and recommend personalized (with respected to preferred slope) routes. The method is based on a weight that integrates user’s history preference, length, and slope.

Assume there are n route options with different lengths and slope variances between a given pair of origin and destination locations, the index value R , which is for evaluating different routes, is calculated as follows:

$$R_k = \frac{l_k}{l_{min}} + \frac{Var_k}{Var_{min}} \times \frac{VarHistory_{current}}{VarHistory_{max}} \tag{4}$$

where l_k is the length of route k ; l_{min} is the length of the shortest route; Var_k is the slope variance of route k ; Var_{min} is the lowest slope variance among all n options; $VarHistory_{current}$ is the average Var of

the user; and $VarHistory_{max}$ is the highest average Var among all the user's trajectories. The lower R value is for a route, the higher probability it will be recommended.

5. Experiment

We evaluated the proposed technique for calculating slopes in real time in this work by conducting an experiment. We did not evaluate the proposed slope network profile technique since its proper evaluation requires PWD subjects willing to travel on different segments with varying slopes, a task which was beyond the scope of this work.

In the experiment conducted, we used existing GPS-based map-matching algorithms and image data to calculate the sidewalk slope. For image processing, we used Matlab, where the RGB of Google Street View images were used as input and the outputs were the edges, the straight lines, and the slope value. The image processing tools utilized in Matlab included `rgb2gray`, `imrotate`, edge functions, and hough functions. The experiment and its result are discussed below.

We randomly selected 100 sidewalk segments in the campus area of the University of Pittsburgh's main campus using PAM's database and searched for 10 images from Google Street View for each segment. The images provided by Google Street View are geo-referenced, each with 800*800 pixels and contains 5–10 meters of the sidewalk according to the width of the road.

In the slope calculation experiment, of the 1000 images used, 117 images resulted in errors below 2 degrees and 52 images resulted in errors below 1 degree, and only 1% of all images produced accurate results. While this is a small number of images with high level of accuracy, we were able to find at least one image for each sidewalk segment of interest in the experiment with an acceptable level of accuracy; 64 out of 100 sidewalk segments have at least one image whose slope calculation error is below 2.0 degrees and 44 out of 100 sidewalk segments have at least one image whose calculation error is below 1.0 degree. Figure 9 shows an example of the calculated slope. After applying the Canny edge detection algorithm and Hough transform, five lines were detected as highlighted in Figure 9c [10–12]. These lines include those representing the sidewalk and the background building. Considering the lowest line representing the sidewalk and the highest line representing the background building, the slope angle is calculated as 8.5 degrees, which is very close to the actual angle 8.42 degrees stored in the PAM database.

Figure 10 shows the overall accuracy of the proposed technique evaluated by comparing the slope calculation result of each image with the actual slope value recorded in PAM's database. The sidewalk accuracy indicates the percentage of images that resulted in accurate slope for the sidewalk. The result shows that 64 segments in total have at least one image that correctly calculated slope (slope error < 1.0 degree), and 33 segments only have 1 image that correctly calculated slope. Although the technique requires only one image to calculate slope, there were cases in which the experiment did not produce accurate results; only 1% of all images in the experiment produced accurate results.

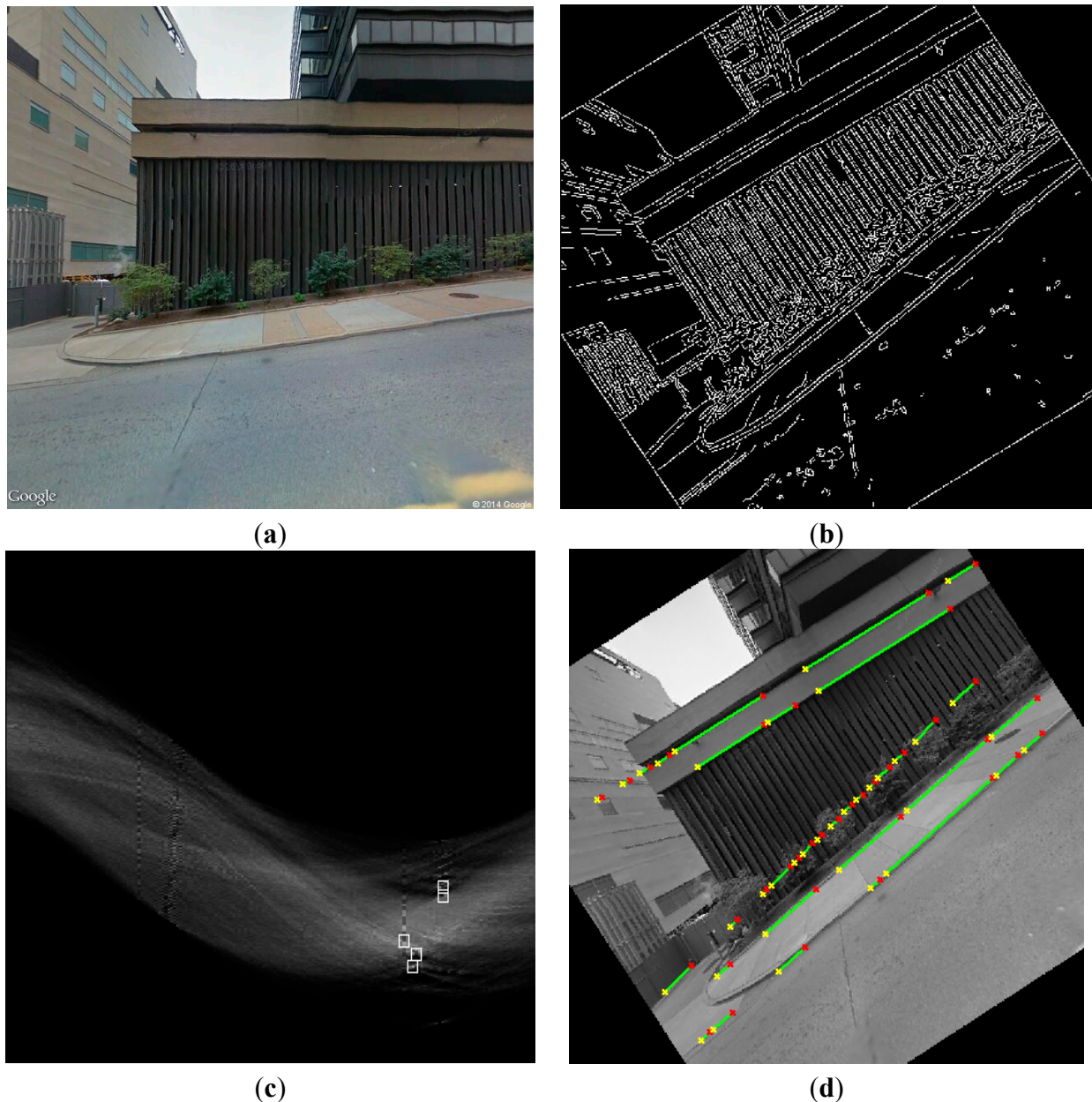


Figure 9. Example sidewalk slope calculation: (a) original image; (b) result of Canny's edge detection algorithm; (c) result of Hough transform, five straight lines detected and shown as light spot circles; (d) straight lines detected in the original image.

Figure 11 shows four examples with incorrect results, each for a different reason. In Figure 11a, the straight lines are detected successfully, however the highest straight line belongs to electric wire which is not suitable for calculating slope. In Figure 11b, due to the surface of sidewalk, which is unrecognizable, the correct sidewalk is not detected. In Figure 11c, there is no building in the background, so slope could not be calculated. In Figure 10d, due to the long distance between the background building and the camera, slope could not be calculated.

These cases above may be addressed by applying different techniques. In one technique, level of confidence in detection, where confidence is high on long continuous straight lines, usually man-made architectures, and on short lines, usually complex backgrounds, can be an option. Another technique is based on voting. As it is not expected that all images result in correct detection, a feasible

technique to avoid incorrect detection is considering multiple images for each sidewalk to calculate slope of the same segment. By analyzing the slope results from all images, the mode of the slope values is considered as the most reliable result, and the frequency of the mode value also indicates the level of confidence on calculation.

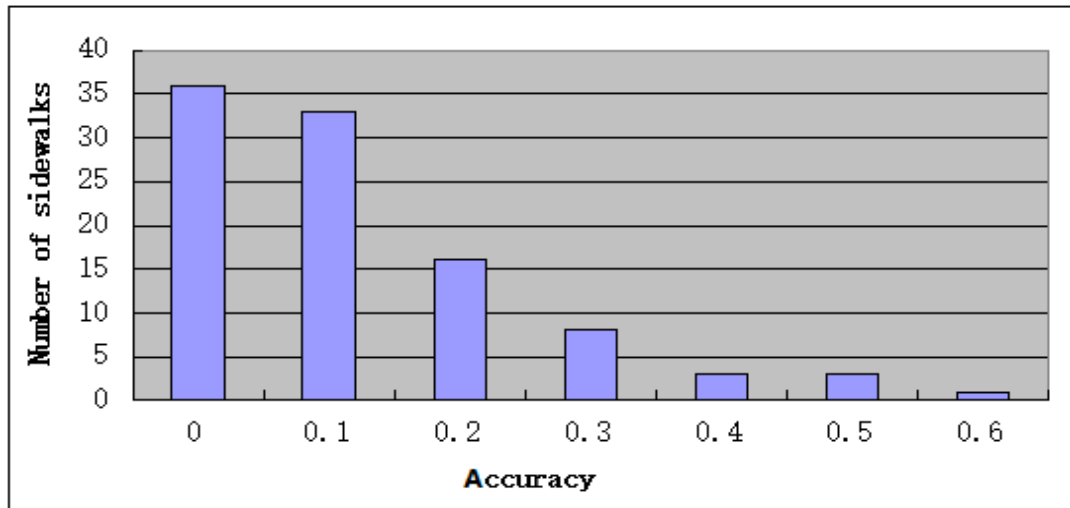


Figure 10. Number of sidewalks and level of accuracy (out of 10 sample images per sidewalk).

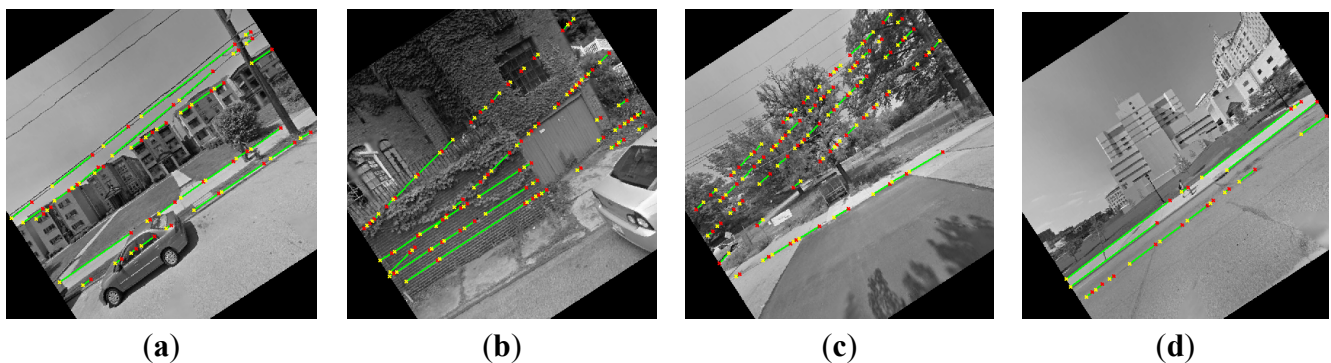


Figure 11. Example cases with incorrect or no slope calculation: (a) unsuitable background data; (b) unclear sidewalk segment; (c) no building in background; (d) background building far away.

6. Conclusion and Future Research

Our technique utilizes GPS trajectory data and Google Street View image data to calculate slope of sidewalks, and recommend suitable routes in real time. An experiment was conducted to evaluate the proposed technique. The experiment result indicates that the technique does not work in all cases and a few cases in which the technique does not work were identified.

While the proposed technique is promising, the work in the paper provided insights into several new research questions. One is about finding a larger number of trajectories and images to test the technique to determine its appropriateness for calculating slopes. Another research question concerns users behavior to further understand their preferences. To address this research question, new image processing techniques to detect additional route features from images, such as sidewalk surface,

material, and obstacles, need to be investigated. Coupling these features with user preferences, navigation services appropriate for assisting PWDs can be developed.

Author Contributions

Yihan Lu: implement the slope calculation algorithm; gather data for the experiment; conduct the experiment; write first draft manuscript. Hassan. A. Karimi: evaluate and correct the slope calculation algorithm; assist with the map-matching algorithm; evaluate the experiment results; revise manuscript drafts.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Thapal, N.; Warner, G.; Drainoni, M.L.; Williams, S.R.; Ditchfield, H.; Wierbicky, J.; Nesathurai, S. A pilot study of functional access to public building and facilities for persons with impairments. *Disabil. Rehabil.* **2004**, *26*, 280–289.
2. Meyers, A.R.; Anderson, J.J.; Miller, D.R.; Shipp, K.; Hoenig, H. Barriers, facilitators, and access for wheelchair users: Substantive and methodologic lessons from a pilot study of environmental effects. *Soc. Sci. Med.* **2002**, *55*, 1435–1446.
3. Smith, V.; Malik, J.; Culler, D. Classification of sidewalks in street view images. In Proceedings of the 2013 International Green Computing Conference (IGCC), Arlington, VA, USA, 27–29 June 2013; pp. 1–6.
4. Senlet, T.; Elgammal, A. Segmentation of occluded sidewalks in satellite images. In Proceedings of the 2012 21st International Conference on Pattern Recognition (ICPR), Tsukuba, Japan, 11–15 November 2012; pp. 805–808.
5. Frackelton, A.; Grossman, A.; Palinginis, E.; Castrillon, F.; Elango, V.; Guensler, R. Measuring walkability: Development of an automated sidewalk quality assessment tool. *Suburb. Sustain.* **2013**, doi:10.5038/2164-0866.1.1.4.
6. Ren, M.; Karimi, H.A. Multisensor map matching for pedestrian and wheelchair navigation. In *Advanced Location-Based Technologies and Services*; CRC Press: Boca Raton, FL, USA, 2013; pp. 209–234.
7. Greenfeld, J.S. Matching GPS observations to locations on a digital map. In Proceedings of the 81st Annual Meeting of the Transportation Research Board, Washington, DC, USA, 13–17 January 2002.
8. Bernstein, D.; Kornhauser, A. *An Introduction to Map Matching for Personal Navigation Assistants*; New Jersey Institute of Technology: Newark, NJ, USA, 1998; p. 16.
9. Karimi, H.A.; Zhang, L.; Benner, J.G. Personalized Accessibility Map (PAM): A novel assisted wayfinding approach for people with disabilities. *Ann. GIS* **2014**, *20*, 99–108.
10. Canny, J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1986**, *8*, 679–698.

11. Ballard, D.H. Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognit.* **1981**, *13*, 111–122.
12. Duda, R.O.; Hart, P.E. Use of the Hough transformation to detect lines and curves in pictures. *Commun. ACM* **1972**, *15*, 11–15.

© 2015 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).