Procedia Computer Science
Volume 80, 2016, Pages 662–672
ICCS 2016. The International Conference on Computational Science

A new Approach for Automatic Detection of Tactile Paving Surfaces in Sidewalks

Marcelo C. Ghilardi, Rafael C. O. Macedo, and Isabel H. Manssour

PUCRS, Faculdade de Informática, Porto Alegre/RS, Brazil
marcelo.ghilardi@acad.pucrs.br
rafael.macedo@acad.pucrs.br
isabel.manssour@pucrs.br

Abstract

In recent years increased the research interest in the development of different approaches to support the mobility of the visually impaired. The automatic detection of tactile paving surface is one important topic of research, not only to help the mobility of visually impaired persons, but also for use in the displacement of autonomous robots, providing a safely route and warnings. In this paper we propose an approach for tactile paving surface detection in real-time with the purpose to assist visually impaired persons. It uses computer vision algorithms combined with decision tree to eliminate some possible false alarms. We assume the visually impaired persons holds a smartphone, which is used to obtain images, as well as to assist him by audio feedback to keep it on the tactile paving surface. This problem is very challenging, mainly due to illumination changes, occlusion, image noise and resolution, as well as different possible colors of the tactile paving surfaces. Experimental results indicate that the proposed approach works well in low resolution images, effectively detecting the tactile paving surfaces in real test scenarios.

Keywords: image processing, tactile paving surface, accessibility application

1 Introduction

The automatic detection of tactile paving surface (also called tactile blocks [11], tactile surface, blister surface or warning surface [3]) is a very important topic of research. The first application that comes in mind is to help in the mobility of visually impaired persons, but it can also be used for the displacement of autonomous robots, providing a safely route and warnings. According to data from the World Health Organization site[1], there are about 285 million visually impaired persons in the world. Among these, 246 million have low vision (less than 30%) and 39 million are blind. Typically, white cane and guide dog are used to help visually impaired persons to walk

[1] http://www.who.int/mediacentre/factsheets/fs282/en

Selection and peer-review under responsibility of the Scientific Programme Committee of ICCS 2016

doi:10.1016/j.procs.2016.05.356

outside [10]. However, even with these resources, mobility autonomy in outdoor environments is a big challenge for these people.

Nowadays, as mentioned in the work of Ahmetovic et al. [1], mobile devices have shown a huge potential in supporting people with disabilities. In addition, since these devices are becoming very accessible, a wide number of assistive technologies have been proposed to support people in everyday activities.

Following this trend, several solutions to aid people with visual impairment are being developed based on images acquired by smartphones. Some examples includes localization of pedestrian crossing [2, 12, 14], stairs [15, 14] and tactile paving surface [17, 6]. However, there are some challenges associated with the use of such images, as the need to correctly point the camera to the target subject, image instability and real time processing. In these solutions, resources as vibration and sound of the device are used to provide feedback to the user.

Considering this context, the main goal of this work is to present a computer vision based approach for the detection of tactile paving surface, which can provide spatial guidance to visually impaired pedestrians. In a nutshell, the proposed approach can provide information about the presence (or absence) and the direction of tactile paving surface near the user. It is initialized by a user command to start the image acquisition step. Then, the images acquired by a camera are used in the crosswalks detection module. The detection is performed by computer vision algorithms that are combined with a decision tree. Finally, the position and direction of the tactile paving surface are detected in relation to the user and a sonorous feedback is provided. The main contributions of the proposed approach are the identification of guidance path and warning surface in three different colors with an accuracy of 88.48%, and the possibility to achieve real time processing.

This paper is structured as follows: In Section 2 we provide an overview of some works for tactile paving surfaces detection. The proposed approach is detailed in Section 3. In Section 4 we present some experimental results. Finally, our conclusions and suggestions for future work are presented in Section 5.

2 Related Work

The shape and the color of tactile paving surface do not have a universal standard, each country or region define their own pattern [9]. For this reason, even with the same goal, in general, many researches developed solutions that can be applied only to some specific region.

The work developed by Jie et al. [6] detects only tactile paving surface with yellow color, specific of a province of China. However, in Brazil, as well as in other countries, there are tactile paving surfaces with different colors. Woo et al. [16, 17] also aim to detect such surfaces for a specific pattern of South Korea, which are very different in color and shape from those used in Brazil.

Another approach presented by Kassim et al. [7] proposes to place RFID tags on all tactile paving surfaces to be able to inform where the user is. This proposal depends on regulation and other bureaucratic obstacles, and has a higher cost, however, countries like China, Japan and Italy have already developed navigation systems for the blind using this technology.

Regarding the input data, Jie et al. [6] get images of the environment using a PDA, and Woo et al. [16, 17] also use images, but without informing the source. Kassim et al. [7] developed a RFID tags detection prototype using components off-the-shelf, as Arduino, smartphone and sensors.

The basic strategy adopted by the models based on images [6, 16, 17] is the tactile paving surfaces detection by targeting color (HSV and HSI) and algorithm edges (kirsch and Canny).

Considering the RFID approach [7], the tag is first detected for a later check in a database to see if there is any information about the location of the tag. Then, verbal feedback about the existence or not of tactile paving surface [6] or about the environment [7] are usually provided.

3 Tactile Paving Surface Detection

Our approach consists of several steps as can be observed in the pipeline presented in Fig. 1. As a first step we need to resize the image in order to get a faster processing. Moreover, if it is in a portrait orientation we need to crop the image. Then, starts the processing to detect the tactile paving surface, which consists of finding and applying several thresholds to extract an area of interest of the image, eliminating dirty and small areas that can not be tactile paving.

After this, occurs the edge detection and the identification of the straight lines related to these edges. The lines are then analysed and the overlapped ones are merged. The area between two lines is analysed to check if it has a pattern acceptable as a tactile paving surface. If so, the resized input image is cropped to get just the region nearest the user. The resultant image is then segmented into several small blocks. Eight different Grey-Level Co-occurrence Matrix (GLCM)[13] are processed for each block (four directions and two distances). The values of entropy, contrast, homogeneity and uniformity are calculated for each GLCM. The results are finally applied to a decision tree that identifies the type of tactile surface. From the type of each of these blocks, the overall type of tactile paving surface is defined. Finally a sound feedback is provided to the user alerting about the type of the tactile paving surface detected and his/her position regarding the center of this surface. A detailed description of each step of this pipeline is available in next subsections.

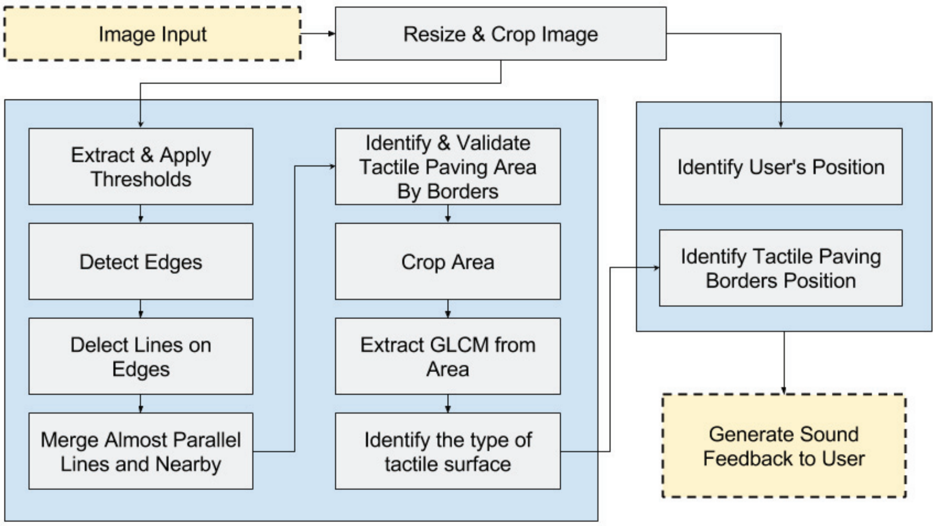


Figure 1: Pipeline of our approach for tactile paving surfaces detection.

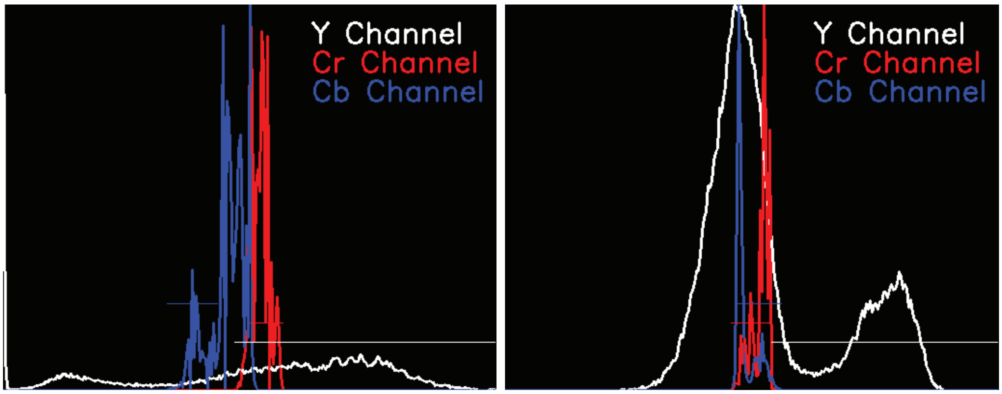
3.1 Area Detection

The implemented approach works with either pre-recorded videos or by direct video input, like webcams. It accepts images in landscape or portrait orientation, however, when the format is portrait, a quarter of the top of the image is excluded since it is not necessary for algorithm. We considered that all images were taken the average height of 1.50 meters and angle measuring 45 degrees relative to the ground.

First, each frame of the video is compressed into a smaller image, which has its width reduced to closer as possible to the value of 300 pixels while maintains its proportion. This compression may cause loss of data, since it merge adjacent pixels into a single one, on the other hand it also reduces the processing time of the algorithms that need to iterate through all points of the image. In order to enable our approach to run near real time, we prefer this gain of speed over data precision. It was chosen the value of 300 pixels for the compression because it was a mean value between speed and precision.

The compressed image is then converted to the YCbCr Channel, or Chrominance, and a histogram for each channel (Y, Cb and Cr) is generated. These histograms are then normalized with the values ranging between 0 to 255. The Chrominance Channel can be used to separate the luma component (Y Channel), or brightness, of an image from its colors, enabling the algorithm to detect and work with shadows. It also help to reduce the overall variance between colors of the same spectrum of light. For example, although Baby Blue and Ultramarine are in the blue spectrum, their variance is big enough to affect image patterns recognition algorithms.

The histogram of the images presents peaks concerning the colors of the tactile paving surfaces. Those peaks combined with threshold methods available in OpenCV[4] are used to determine the possible tactile paving surface area in the image. Since it has different colors, we defined two configurations to represent their areas in the histogram. Those configurations were here called *Type A* and *Type B*. *Type A* works better with brighter colors as yellow or blue, and *Type B* works better with opaque colors as gray. This process generate a total of 6 thresholds values for the 3 channels of the image (Min/Max Y, Min/Max Cr and Min/Max Cb). Those thresholds, when applied to the image, can delimit the possible tactile paving surface area.



(a) Type A Histogram (b) Type B Histogram

Figure 2: Histograms of tactile paving surfaces.

The used threshold values can change, since they are obtained through formulas empirically defined. All the fixed threshold values presented in the processes were the best threshold values found after using several heuristics. The formulas can be observed on Table 1.

Table 1: Thresholds Formulas

Type	Threshold	Formula
A	Min Cb	No Threshold
	Max Cb	Between the first point in the Cb histogram with value greater than or equal 40 and the histogram highest peak, the threshold is the last point whose value is less than or equal 40 before a high peak (value greater than or equal 200)
	Min Cr	From the highest peak to the first point of the Cr histogram, the threshold is the first point with value greater than the value found at the highest peak in the Cb histogram or the first point with value below 135
	Max Cr	No Threshold
	Min Y	No Threshold
	Max Y	No Threshold
B	Min Cb	From the highest peak to the first point of the Cb histogram, the threshold is the first point with value less than or equal 40
	Max Cb	No Threshold
	Min Cr	No Threshold
	Max Cr	After the highest peak of the Cr histogram, the threshold is the first point with value less than or equal 135 or the last point in this value range if a high peak is found (value greater than or equal 200)
	Min Y	After the highest peak in the Cr histogram, the threshold is the first point with value less than or equal 50.
	Max Y	No Threshold

After applying the thresholds, when the image has tactile paving surface, erosion and dilation morphology operators are applied to eliminate possibles false positives. Otherwise, if the amount of pixels detected as an interest area is less than 10% or more than 40% of the image, it is defined as image without tactile paving surface. An example of the area detected by these processes can be observed in Fig. 3.

(a) Original (b) Type A (c) Type B

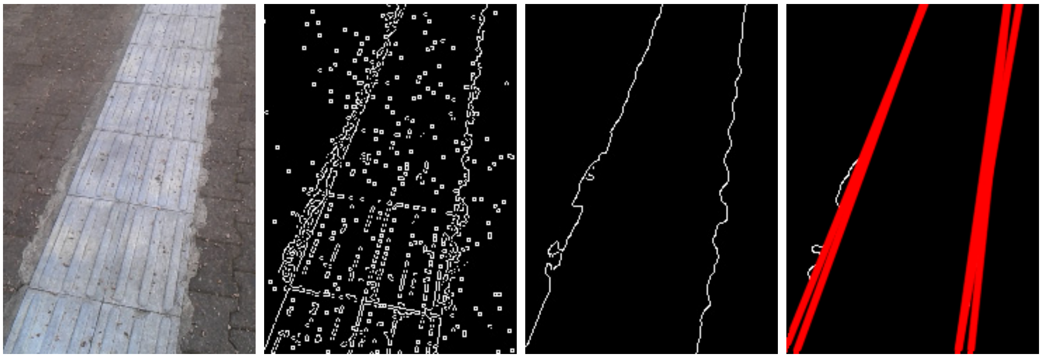
Figure 3: Example of a detected area using configuration *Type A* and *Type B*.

Upon calculating the thresholds from *Type A* or *Type B*, the possible area of the tactile paving is extracted from the image. The usage of *Type A* or *Type B* in the algorithm is determined by the lack of detection of "parallel lines" in the image. In this paper, we considered as "parallel lines" the border lines of the tactile paving surface that point to the same direction but are not necessarily equidistant, being able to cross at some point. The algorithm starts using *Type A* detection and if in one frame, no "parallel lines" are detected, then the last detected lines will be used. However, if the frame in which these lines were detected is 3 or more frames before the current frame then the algorithm will change the type of detection. This feature aims to keep the tactile paving surface detection as constant as possible without allowing the user to walk on possible false positives for too long.

3.2 Borders Detection

In order to verify if the detected area contains a tactile paving surface, it is possible to check for the "parallel lines", as theses surfaces are usually long straight lines. One solution to detect these "parallel lines" is to use the Canny edge detector[5].

Since the detected lines need to be smooth, and it is necessary to remove the noise, a blurring filter (median blur) is applied to the image before running the edge detection algorithm. Then the Hough Lines Transform[5] is used on the resultant image. The results of these steps can be seen in Fig. 4.



(a) Original Image (b) Canny (c) Canny and Blur (d) Hough Lines

Figure 4: Algorithms for detecting the edges.

Its worth noticing that each detected line would need to be analysed later in our algorithm, decreasing its performance. On the other hand, when using high thresholds levels, irregulars tactile pavings surfaces, like dirty or highly segmented tactile pavings, stop being detected. Considering obstacles along the way, abrupt turns or even a little deviation of the user in relation to the tactile paving surface, and after a study of the threshold levels, the fixed threshold level of 25 proved to be able to detect short tactile pavings segments without many false positives. This value was then chosen as the threshold level of the Hough Lines Transform.

The detected lines, however, are often overlapped and represent the same line in the original image, but with just slightly different angles. Then, to have better performance and result, we implemented a function that merges the detected lines that are close together and have similar angles.

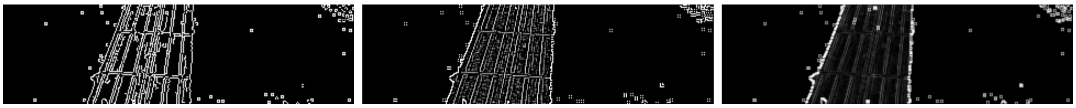
Sometimes, even after merging the detected lines, there are still more than two lines in the results. In this case, we choose the two most vertical lines that are also "parallel" to each other and whose distance specifies an area compatible with a tactile paving surface (i.e. not too large or too thin). We check the theta angle on each line and the lines whose theta differ between 0.10 and 0.45 are considered "parallel lines". These values were chosen after testing several different values. It is also checked the direction of lines, i.e., in the base of the image (near the user) the distance between the lines need to be higher than in the top of the image.

Finally the pair of "parallel lines" with the lowest values of theta and that are compatible in size with a tactile paving surface, are then selected as the border of the tactile paving. Although this method still produces some false positives, it is the best method found compared to other solutions as, e.g., simply selecting the two most "parallel lines" or applying a flood fill to check if the resulting polygon would match one tactile paving surface. After detected the borders of the tactile paving it becomes easy to calculate its middle point.

3.3 Validation Testing

The detection of the "parallel lines" does not ensure that they are the borders of a tactile paving surface and not another similar surface. After testing this implementation on several videos and images it became clear that the image need to be further analysed. Thus, the image is segmented into blocks of 25 pixels by 25 pixels that are numerated and the GLCMs are calculated for each one in eight different directions. The results of the GLCM depend on the offset, or direction, specified. We chose to use eight different offsets in our approach, resulting in eight different matrices.

Although the Canny edge detector was used before applying the Hough Lines Transform, it is not suitable for the detection of small variations of the edges. After the analysis of several results it was chosen to use the Laplace operator to detect edges before applying the GLCM. A comparison of the application of Canny, Laplace and Sobel methods can be observed on Fig. 5.



(a) Canny (b) Laplace (c) Sobel

Figure 5: Comparison among different edge detector for image segmentation.

To improve the performance, just the lower half of the image is processed, since the other half is usually too far away for a precise calculation of the GLCM. Additionally, to also improve the performance, blocks that have more than 75% of black pixels are discarded (blank blocks). The resultant block division can be observed on Fig. 6, where green blocks are blocks that will be analysed and blue blocks are discarded.

For each generated GLCM, in each block, we extract the values of entropy, contrast, homogeneity and uniformity (or energy). A spreadsheet containing 624 entries from blocks was created, and for each entry all values were stored for each of the four directions and two distances. Using a data miner, RapidMiner[2], we extracted a decision tree from these values that allow to predict the type of the block (Alert, Directional or Noise). The Alert type indicates an alert tactile paving surface, the Directional type indicates a directional one, and the Noise type is everything else. In order to determine the type of the image we used the following rules:

- **Alert:** If two neighbors blocks of the up, right, left or down directions are of type Alert, then the image (or frame) is considered to have an alert tactile paving surface;

[2]https://rapidminer.com/products/studio/

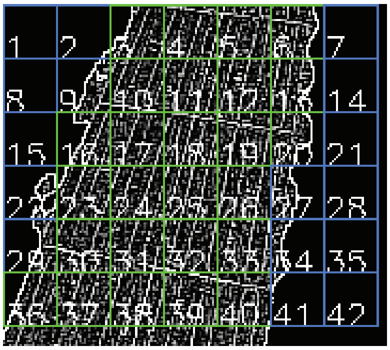


Figure 6: Block segmentation.

- **Directional:** If the first rule does not apply and a square of size 2x2 of blocks of type Directional is found, then the image (or frame) is considered to have a directional tactile paving;
- **Noise:** If none of the two previous rules apply, then the image is considered as a noise (no tactile paving detected).

These rules help to reduce the amount of false positives occurrences while preserving the quality of the detection of real tactile paving surfaces. An image example can be observed in Fig. 7a and its resulting blocks division with the detection of an alert tactile paving surface can be observed in Fig. 7b.

The second rule to indicate a directional tactile paving surface, aims to eliminate the majority of false positives while keeps detecting even thin or small tactile paving surfaces. As directional tactile paving surfaces are composed by straight lines, it is possible to confirm its detection if a small, but complete, group of 2x2 blocks is found in the image. An example of this can be observed on Fig. 7c. The resulting division of the blocks for this image can be observed in Fig. 7d.

(a) Alert Paving (b) Alert Paving (c) Directional Paving (d) Directional Paving

Figure 7: Example of an alert and directional tactile paving surface.

4 Experimental Results

In this section we introduce the experiments done and the reached results. We start describing how the images were obtained and the ground truth was created for evaluation. After, we present the evaluation of our approach for tactile paving surfaces detection.

All the experiments were done in a notebook with a Intel Core i5 processor and 4GB of memory. For implementation, we used C++ programming language, OpenCV[4] library version 3.0 and OpenAL[8] library for the sound feedback.

4.1 Ground Truth

Images of several sidewalks with different lighting conditions were acquired and used to create the ground truth (GT). All these images were obtained through a camera attached to the user's abdomen, positioned at an average height of 1.0 meter and with a 45 ° angle relative to the ground, as shown in Fig. 8. In this way, the user can keep his hands free.

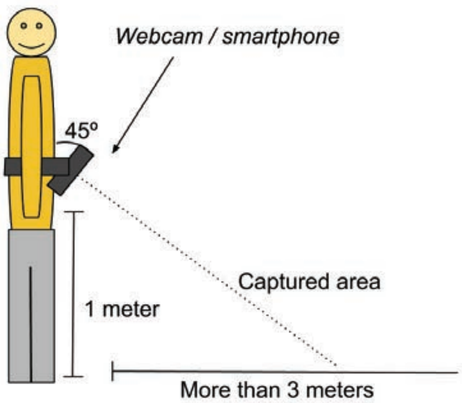


Figure 8: Image capture position.

A total of 521 images compose the GT: 320 are sidewalks with tactile paving surfaces and 201 are sidewalks without it. This means that for each original image, there is an image at the GT with the edges of the tactile paving surface, when present, delineated by two red lines, as shown in Fig. 9. The evaluation algorithm gets the lines of the GT image and compares with the lines generated by our approach to determine the accuracy.

(a) Original Image (b) GT Image

Figure 9: Example of an original and GT image.

4.2 Evaluation

The evaluation results are shown in table 2, where TP, TN, FP, and FN refer to True Positive, True Negative, False Positive and False Negative, respectively. The numbers associated with

TP and TN correspond, respectively, to images with and without tactile paving surfaces that were correctly identified. FN and FP correspond, respectively, to images with and without tactile paving surfaces that were erroneously predicted by our approach.

Table 2: Evaluation Results

		Ground Truth	
		Tactile Paving	No Tactile
Predited by model	Tactile Paving	273 (TP)	13 (FP)
	No Tactile	47 (FN)	188 (TN)

We used Accuracy, Sensitivity and Specificity as evaluation measures, whose formulas and results are shown in table 3. The specificity of 93.53% indicates that images without tactile paving surfaces are correctly predicted in most cases. We consider the specificity important for user safety, since it indicates less false positives. The sensitivity result of 85.31% and the accuracy of 88.48% demonstrates the overall efficiency of the presented approach. The processing rate was separately measured through the execution of the detection process on 7715 frames sample and the extraction of the average processing time. The obtained processing rate was 16.27 fps, approximately.

Table 3: Rate Results

Measure		**Rate**
Accuracy	(TP+TN)/(P+N)	88.48%
Sensitivity	TP/P	85.31%
Specificity	TN/N	93.53%

5 Conclusions

In this work we proposed a new approach for tactile paving surface detection with the purpose to assist visually impaired persons. The images used for processing can be acquired by the user through his smartphone or camera fixed to his body. Then, computer vision algorithms are combined with decision tree to provide information about the presence (or absence) and direction of tactile paving surface near the user.

Despite the challenges of illumination changes, occlusion, image noise and resolution, experimental results indicated that the implemented approach effectively detect the tactile paving surface achieving about 88.48% of accuracy. Then, an audio feedback assist the user to keep in the tactile paving surface.

For future work we intend to embed this approach to a smartphone and match with obstacle detection models, in order to increase the tactile paving surface detection accuracy and the safety. We also want to develop a case study with visually impaired people to evaluate the real applicability of the proposed approach.

Acknowledgment

Authors would like to thank CNPq, Hewlett-Packard Brasil Ltda. and PUCRS for the financial support. The authors also thank the collaboration of the student Lucas Pedreira da Silveira.

References

[1] D. Ahmetovic, C. Bernareggi, A. Gerino, and S. Mascetti. Zebrarecognizer: Efficient and precise localization of pedestrian crossings. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 2566–2571, Aug 2014.

[2] T. Asami and K. Ohnishi. Crosswalk location, direction and pedestrian signal state extraction system for assisting the expedition of person with impaired vision. In *Mecatronics (MECATRONICS), 2014 10th France-Japan/ 8th Europe-Asia Congress on*, pages 285–290, Nov 2014.

[3] Department for Transport. *Guidance on the use of tactile paving surfaces*. GOV.UK, 2007.

[4] G.B. García, O.D. Suarez, J.L.E. Aranda, J.S. Tercero, I.S. Gracia, and N.V. Enano. *Learning Image Processing with OpenCV*. Community experience distilled. Packt Publishing, 2015.

[5] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Pearson Education, 2011.

[6] Xu Jie, Wang Xiaochi, and Fang Zhigang. Research and implementation of blind sidewalk detection in portable eta system. In *Information Technology and Applications (IFITA), 2010 International Forum on*, volume 2, pages 431–434, July 2010.

[7] AM. Kassim, H.I Jaafar, M.A Azam, N. Abas, and T. Yasuno. Design and development of navigation system by using rfid technology. In *System Engineering and Technology (ICSET), 2013 IEEE 3rd International Conference on*, pages 258–262, Aug 2013.

[8] E. Lengyel. *The OpenAL Programming Guide*. Charles River Media Game Development. Charles River Media, 2005.

[9] Jiangyan Lu, K.W.M. Siu, and Ping Xu. A comparative study of tactile paving design standards in different countries. In *Computer-Aided Industrial Design and Conceptual Design, 2008. CAID/CD 2008. 9th International Conference on*, pages 753–758, Nov 2008.

[10] V.N. Murali and J.M. Coughlan. Smartphone-based crosswalk detection and localization for visually impaired pedestrians. In *Multimedia and Expo Workshops (ICMEW), 2013 IEEE International Conference on*, pages 1–7, July 2013.

[11] Rikuo Sakaguchi, Shino Takasu, and Tetsuo Akiyana. Study concerning the colors of tactile blocks for thr visually handcapped –visibility for the visually handicapped and scenic congruence for those with ordinaty sight and vision–. In *JIPEA World Congress*, page 453462, Tokio, Japan, 2000. ACM.

[12] Longfei Shangguan, Zheng Yang, Zimu Zhou, Xiaolong Zheng, Chenshu Wu, and Yunhao Liu. Crossnavi: Enabling real-time crossroad navigation for the blind with commodity phones. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '14, pages 787–798, New York, NY, USA, 2014. ACM.

[13] L.G. Shapiro and G.C. Stockman. *Computer Vision*. Prentice Hall, 2001.

[14] Shuihua Wang, Hangrong Pan, Chenyang Zhang, and Yingli Tian. Rgb-d image-based detection of stairs, pedestrian crosswalks and traffic signs. *Journal of Visual Communication and Image Representation*, 25(2):263 – 272, 2014.

[15] Shuihua Wang and Yingli Tian. Detecting stairs and pedestrian crosswalks for the blind by rgbd camera. In *Bioinformatics and Biomedicine Workshops (BIBMW), 2012 IEEE International Conference on*, pages 732–739, Oct 2012.

[16] Byung-Seok Woo, Sung-Min Yang, and Kang-Hyun Jo. Brick path detection from shape pattern and texture feature. In *System Integration (SII), 2011 IEEE/SICE International Symposium on*, pages 78–83, Dec 2011.

[17] Byung-Seok Woo, Sung-Min Yang, A. Vavilin, and Kang-Hyun Jo. Brickpath region detection using color and shape pattern information. In *Strategic Technology (IFOST), 2011 6th International Forum on*, volume 2, pages 720–724, Aug 2011.