Real-time Multiresolution Crosswalk Detection with Walk Light Recognition for the Blind

Krešimir ROMIĆ, Irena GALIĆ, Hrvoje LEVENTIĆ, Krešimir NENADIĆ
University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology, Kneza Trpimira 2B, HR-31000 Osijek, Croatia
kromic@etfos.hr

Abstract—Real-time image processing and object detection techniques have a great potential to be applied in digital assistive tools for the blind and visually impaired persons. In this paper, algorithm for crosswalk detection and walk light recognition is proposed with the main aim to help blind person when crossing the road. The proposed algorithm is optimized to work in real-time on portable devices using standard cameras. Images captured by camera are processed while person is moving and decision about detected crosswalk is provided as an output along with the information about walk light if one is present. Crosswalk detection method is based on multiresolution morphological image processing, while the walk light recognition is performed by proposed 6-stage algorithm. The main contributions of this paper are accurate crosswalk detection with small processing time due to multiresolution processing and the recognition of the walk lights covering only small amount of pixels in image. The experiment is conducted using images from video sequences captured in realistic situations on crossings. The results show 98.3% correct crosswalk detections and 89.5% correct walk lights recognition with average processing speed of about 16 frames per second.

***Index Terms*—assistive technology, image recognition, machine vision, morphological operations, object detection.**

I. INTRODUCTION

One of the main challenges that the blind and visually impaired people meet in their everyday life is safe and independent road crossing. Electronic aid systems for helping the blind in their movement are becoming increasingly common. Such systems appear in a variety of forms [1] where camera based systems takes a large part. By the survey [2-3], wearable camera based systems belongs to the group of electronic travel aids (ETA) which represents systems that provide information about the environment which is normally acquired through vision. In this work, we propose image processing algorithms as a part of camera based aid for the blind. Developed algorithms provide useful information about crosswalk and walk light to help the user when crossing the road. Aim is to detect crosswalks from input images captured by camera in motion. It is supposed that camera will be positioned in the chest height and that the possible pedestrian crosswalk will be in the camera field of view. Output of the algorithm is the decision about presence of the crosswalk and additional information about walk light sign. The way of providing the feedback for the blind person is not in the focus of this work, but it is often provided through audio or tactile signals [2].

In this paper, accent is on developing high accuracy method for detection of crosswalks and walk lights which will be possible to run on portable computer using simple and cheap monocular camera. Other than the simplicity and affordability of such system components, method should be expandable with other functionalities.

Crosswalk detection algorithms are often considered in the context of vehicle safety systems or autonomous vehicles [4]. Application of such algorithms in electronic aids for the blind and visually impaired is not common. It is important to perceive that the position and orientation of the crosswalk is different when looking at it from the pedestrian perspective. Therefore, approaches for helping the blind pedestrians mostly differ from those in vehicles.

Camera based approaches for crosswalk detection for the blind and visually impaired can be roughly divided in two categories according to camera type. Some authors use monocular cameras while others prefer stereo cameras or similar devices capable of capturing the depth information. Use of simple and cheap monocular camera is proposed in [5] where template matching and normalized correlation method is used for detection. Authors in [6] use color-based preprocessing, mean shift segmentation and morphological post-processing. Another approach for monocular camera is proposed by Uddin and Shioyama [7] where bipolarity-based segmentation is proposed. Several publications about this subject were published as part of the Crosswatch project led by James Coughlan. Figure-ground segmentation is proposed in [8] and their work is specialized for usage on smartphones where additional information is gained by GPS [9]. When analyzing methods that use the depth information from RGB-D image, work of S. Wang stands out with Hough transform and depth features [10]. Novel approach in [11] uses deep learning techniques on RGB-D images[11].

Algorithms for detection and recognition of traffic lights are mostly targeted on traffic lights for vehicles like in [4], [12], while smaller number of authors deal with pedestrian walk lights [10], [13-14]. One thing that is common to both is color based approach which extract red or green color regions at first place and then matches the shape of the light sign. Walk sign style may differ in various countries and continents. In this paper, we are dealing with signs characteristic for European countries and our algorithm is focused on recognition based on small resolution images where light sign covers only few pixels. Our method for walk light recognition relies on previous crosswalk detection method and those methods are conceived to be a part of unified system. Authors in [10] and [13] have the similar approach, while those in [14] are solving walk light recognition as a separate problem on images with higher resolution.

This paper is structured as follows: second chapter

Digital Object Identifier 10.4316/AECE.2018.01002

explains proposed multiresolution approach, third chapter explains crosswalk detection algorithm, fourth chapter deals with pedestrian walk lights recognition, and fifth chapter presents obtained experimental results.

II. MULTIRESOLUTION APPROACH

In order to gain better processing speed, we propose the processing of downscaled images in particular algorithm steps. This chapter explains the sequence of those algorithm steps and benefits gained by using rescaled input images.

Proposed algorithm for crosswalk detection roughly can be decomposed into preprocessing and detection part. Both parts consist of several steps as presented on diagram in Fig. 1. At the very beginning, image is grabbed from video as an input data. Such images are grabbed one after another and the goal is to process as much images as possible and to provide information about crosswalk on every processed image. After determining existence of crosswalk, additional part of the algorithm is performed to recognize red or green walk light if one is present. This part of the algorithm is performed only if crosswalk is detected on that particular image. As an output of the algorithm information about detected crosswalk and walk light are expected.

Proposed method is tailored to work with input images in resolution of 640×360 pixels which proved to be sufficient to detect objects like crosswalks from distance of 1 to 3 meters [15]. However, it is possible to perform particular steps in lower resolution which significantly speeds up processing.

First part of the method which is performed on downscaled image is preprocessing. Input image is downscaled to 80×45 resolution and regions of interest are localized in this step. Further processing of localized regions of interest is performed again in initial resolution thereby avoiding the processing of unnecessary pixels outside the regions of interest. Another downscaling of images is performed when line energy is calculated based on edge-points mapped on downscaled images. Light recognition process is performed on initial image in 640×360 resolution and final decision is provided.

This rescaling lifecycle of the image is showed on Fig. 2. According to the look of downscaling and upscaling sequence as shown in Fig. 2, this approach can be called W-cycle multiresolution algorithm similar to W-cycle multigrid algorithms for solving differential equations [16].

III. CROSSWALK DETECTION

Method for real-time pedestrian crosswalk detection in video sequences is the central part of the proposed algorithm. This chapter explains in detail all techniques developed and used in the detection process. Detection process consists of the following steps: preprocessing in low scale, morphological analysis and potential edge-points detection, line energy and decision making.

In the preprocessing step input image is being prepared for finding regions of interest. Since the crosswalk stripes on the road are mostly in white color, regions of interest will be sought as larger white regions. However, those white crosswalk regions should be visible on images with smaller resolution.

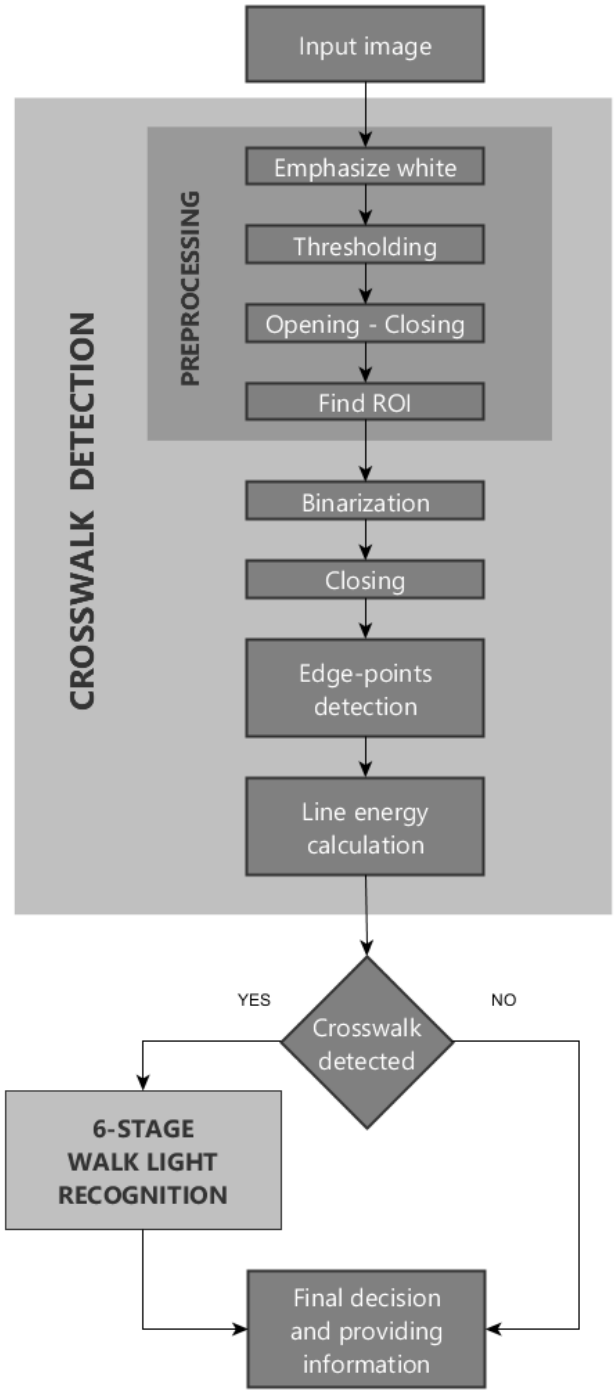


Figure 1. Proposed algorithm steps

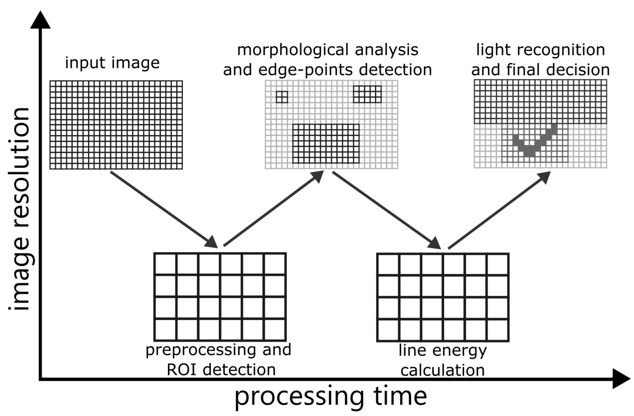


Figure 2. W-cycle resolution rescaling sequence

In order to reduce processing time, all preprocessing actions are performed on downscaled images in resolution 80×45 which is eight times smaller in width and eight times smaller in height. Example of input image in 640×360 resolution is shown in Fig. 3 a) and downscaled image is shown in Fig. 3 b).

a)

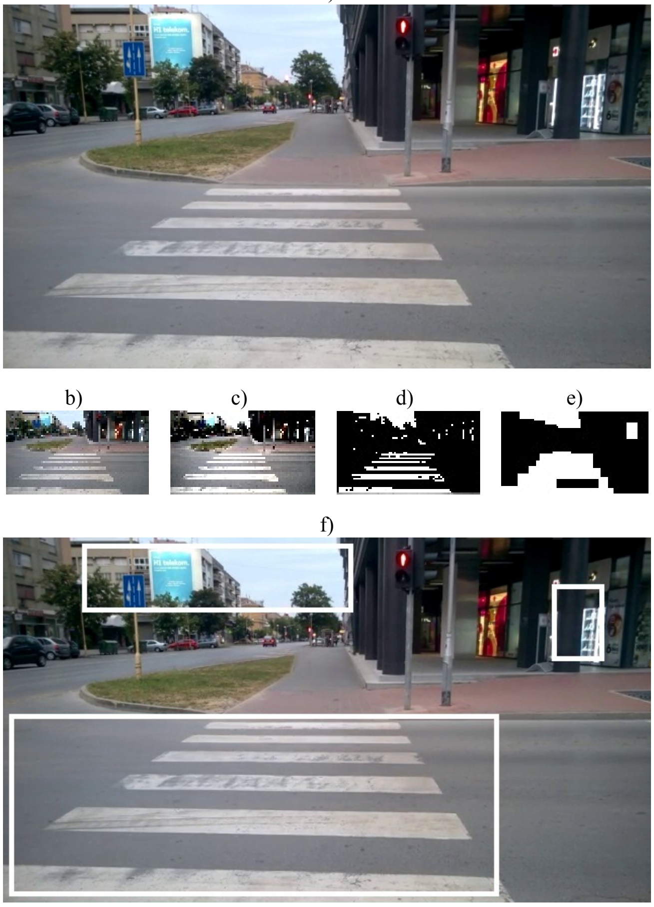


Figure 3. Preprocessing: a) input color image, b) downscaled image, c) emphasized white color, d) binary image, e) binary image after opening and closing, f) detected regions of interest

First preprocessing action is emphasizing nearly white colors in image because digital representation of white color often differs from our human perception [17]. This is solved by changing the red, green and blue channel values of pixels in color image according to equation (1) where new value for red color channel is $f^*_R(x,y)$.

$$f^*_R(x,y) = f_R(x,y) - (avg_R - g(x,y)) \quad (1)$$

where $f_R(x,y)$ is red channel value in x-th row and y-th column, avg_R is average red channel value for the whole image. Gray value of observed pixel $g(x,y)$ is calculated as weighted sums of three channels [18] as shown in (2).

$$g(x,y) = 0.299 * f_R(x,y) + 0.587 * f_G(x,y) + 0.114 * f_B(x,y) \quad (2)$$

This process is repeated for green and blue channels on every pixel in image and the final result is visible in Fig. 3 c). The process of emphasizing the white color compared to other colors enhances contrast in image, especially in crosswalk region due to white stripes. This proved to be more useful for this purpose than some contrast enhancement techniques such as histogram equalization. Although the method is primarily developed to work on day scenes, previous step of emphasizing the white colors allows the method to work with low contrasted images taken in the evening or early in the morning when lighting conditions are not ideal (Fig. 4). This also refers to night images taken under infrared light source (Fig. 5).

Once we have emphasized white color, downscaled color image is converted to binary image. Thresholding is performed to set every pixel with all three channel values larger than 190 to white color. Thereby, additional condition is checked to exclude pixels with difference between two channel values larger than 20. Obtained binary image is shown in Fig. 3 d).

As a final part of preprocessing, two iteration of morphological closing operation is performed on binary image accompanied by two iterations of opening operation. Structuring element 3×3 is chosen for both operations. This closing and opening operations help to exclude small white areas and fill larger white areas to form a consistent shape as shown in Fig. 3 e) [19]. All remaining shapes are potential crosswalk regions and rectangles around them represent the regions of our interest (ROI). Positions and sizes of those regions are now transferred to original image in 640×360 resolution and further processing is performed exclusively on those regions of interest (Fig. 3 f).

Multiple rectangular regions of interest are possible to obtain in the previous step. All regions were further processed as potential crosswalk regions. If there are no regions of interest, further processing is not necessary and algorithm proceeds to the next image.

a)

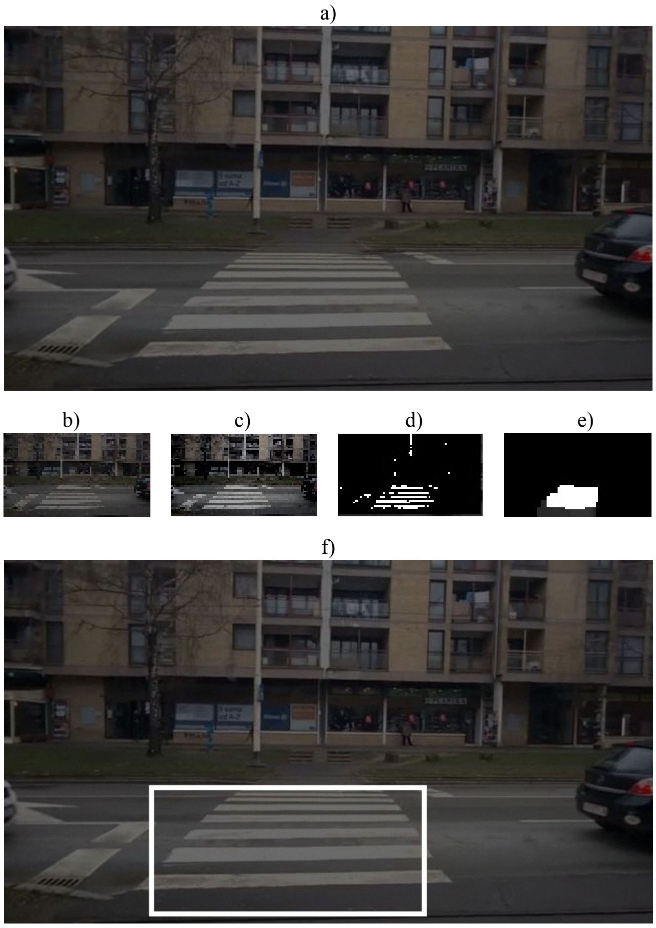


Figure 4. Preprocessing on low-contrast image: a) input color image, b) downscaled image, c) emphasized white color, d) binary image, e) binary image after opening and closing, f) detected regions of interest

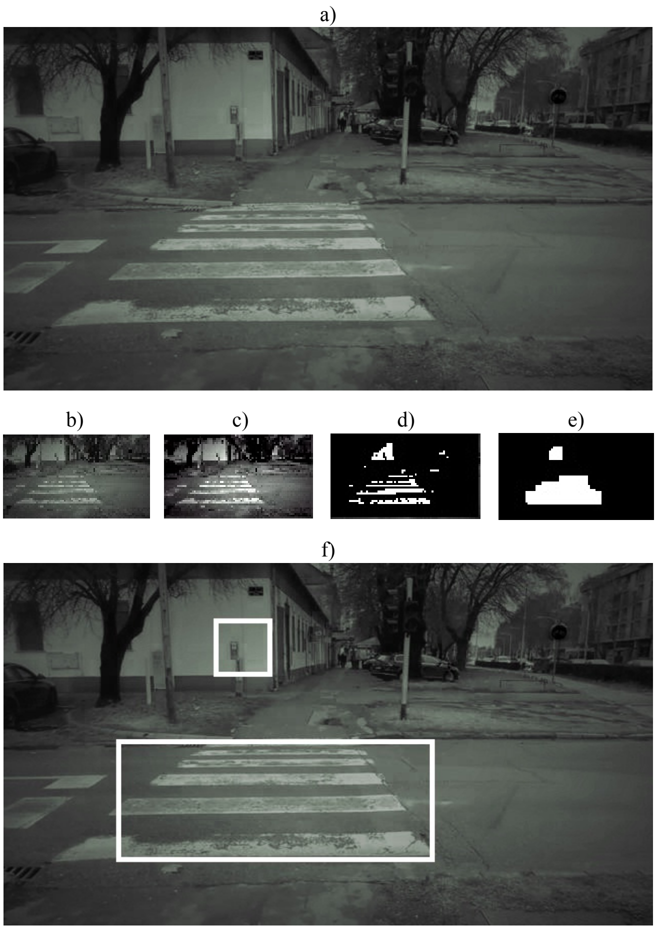


Figure 5. Preprocessing on simulated night scene with infrared light source: a) input color image, b) downscaled image, c) emphasized white color, d) binary image, e) binary image after opening and closing, f) detected regions of interest

Every ROI is processed and analyzed to distinguish whether it contains crosswalk or not. First step is to convert color values to binary values based on experimentally obtained threshold value of 190 for every channel. After thresholding, binary image looks similar to example on Fig. 6 a). Obtained binary image carries enough information about the lighter and whiter objects such as crosswalk but also minimizes unnecessary data in image. Next additional step is morphological closing operation which is here performed using 5×5 structuring element. This operation fills small black gaps in the image. When talking about crosswalk region, such cases are caused by the washed out white paint or stains on crosswalk stripes. Result of this step is shown in Fig. 6 b).

Figure 6. Regions of interest after: a) thresholding, b) closing operation

Obtained binary representation of ROI is forwarded to further analysis which has a main goal to detect specific points that represent crosswalk stripe edges. Such points will be called edge-points. Method for detecting edge-points is based on the increasing vertical sequences of the white or black pixels characteristic for the crosswalk region similar as proposed in [20]. Every vertical transition from white to black is marked as edge-point if the length of white pixel streak increased from previous by 20% to 110%. The process is similar for vertical transitions from black to white where the lengths of consecutive black pixel streaks are compared. This process is visualized on Fig. 7 a). After the analysis of every column in ROI, detected crosswalk edge points look similar as shown on example in Fig. 7 b). This figure shows edge-points detected in all three regions of interest.

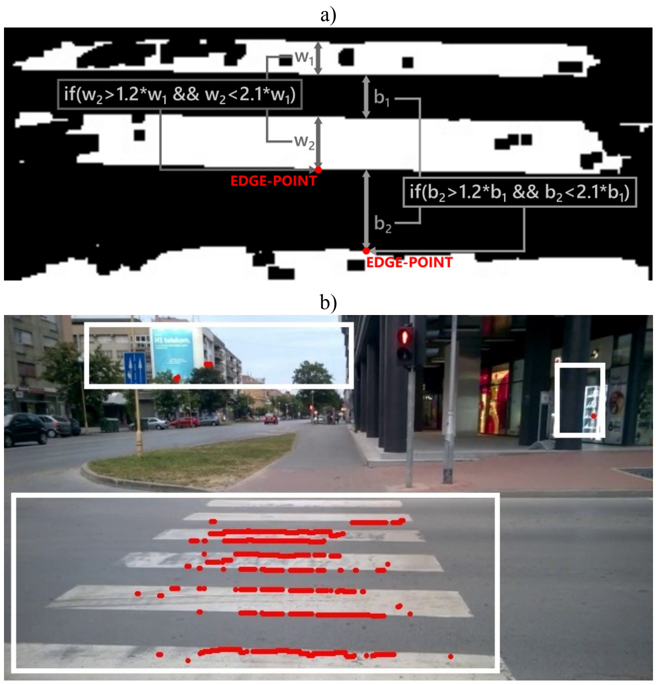


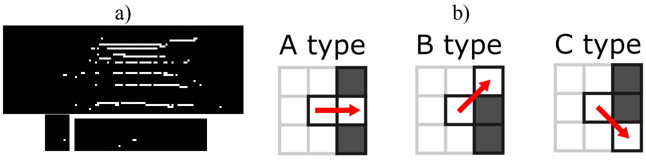
Figure 7. a) Edge-points detection conditions, b) Detected edge-points in every ROI

Edge-points detected in previous step are used to make final decision whether they belong to crosswalk or not. Coordinates of those edge-points are mapped to low scale blank image which has width and height 4 times smaller than the ROI. Rules for mapping the edge-points on low-scale image are given in (3):

$$b(x/4, y/4) = \begin{cases} 255, & f(x,y) \in EP \\ 0, & f(x,y) \notin EP \end{cases} \quad (3)$$

where b is the new value of pixel for low-scale image, x and y are coordinates of the pixel in ROI, f is the pixel value in ROI and EP is the set of pixels that represents edge-points.

This is the second processing on downscaled images in this algorithm and it should highlight horizontal lines that represent crosswalk edges. This downscaled image is given in Fig. 8 a). Since the edge-points are now scattered to a smaller area, they form nearly horizontal lines which are often continuous in the crosswalk region. In order to find out if the mapped points belong to crosswalk region one parameter called line energy is calculated. Line energy is calculated by finding the groups of connected white pixels in image and passing through all connected white pixels from left to right. In this process three right neighboring pixels are observed for every point and there are three possible types of transition to the right side as shown in Fig. 8 b). When meeting one of those transitions, counter for it is incremented and added to the overall sum. The sum will be higher if there are more consecutive white pixels in the same direction. When there is no more consecutive white pixel on the right side, algorithm moves to another white pixel group in image. Finally, the line energy is calculated as the ratio of the aforementioned sum and the number of the white pixels in image. This process is described with pseudo-code given in Fig. 8 c). If there are more nearly horizontal lines, line energy is getting higher values. Otherwise, if there are only white dots without lines, line energy will strive towards 1. Experimental results proved that images with line energy smaller than 2 do not contain crosswalk. On the other hand, when the line energy is 2 or higher, crosswalk is detected. The final decision is made based on that parameter and Fig. 9 shows obtained line energy values for every image with and without crosswalk in the dataset of 240 images in total.



c)

```
n=0
For every group of white pixels
        flag = 0
        A = B = C = 0
        n++
        While (flag == 0)
                n++
                If (A type)
                        A++
                        Sum = Sum + A
                        Move to right
                Else if (B type)
                        B++
                        Sum = Sum + B
                        Move to up-right
                Else if (C type)
                        C++
                        Sum = Sum + C
                        Move to down-right
                Else
                        flag = 1
LineEnergy = Sum/n
```

Figure 8. a) Low-scale binary image with all edge-points, b) Types of transitions, c) Pseudo-code for line energy calculation

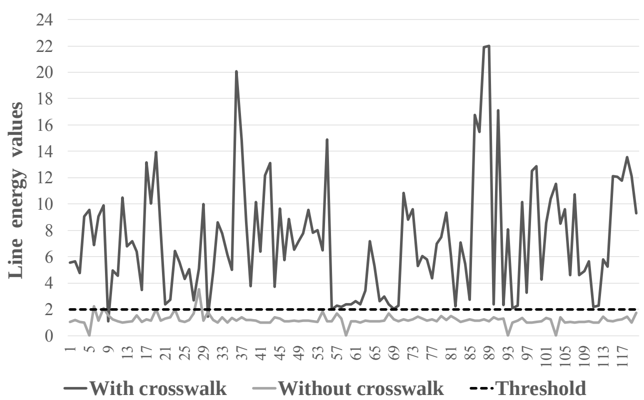


Figure 9. Line energy among dataset of images with and without crosswalk

Once the ROI with crosswalk is detected, correct direction for the blind person can be determined. The correct direction is calculated as the angle between the vertical line which passes through center point of the ROI and line from the bottom middle point of the image to the center point of the ROI. To simplify the information about direction for the blind person, angle size is mapped to 5 possible ways as shown in Fig. 10 a).

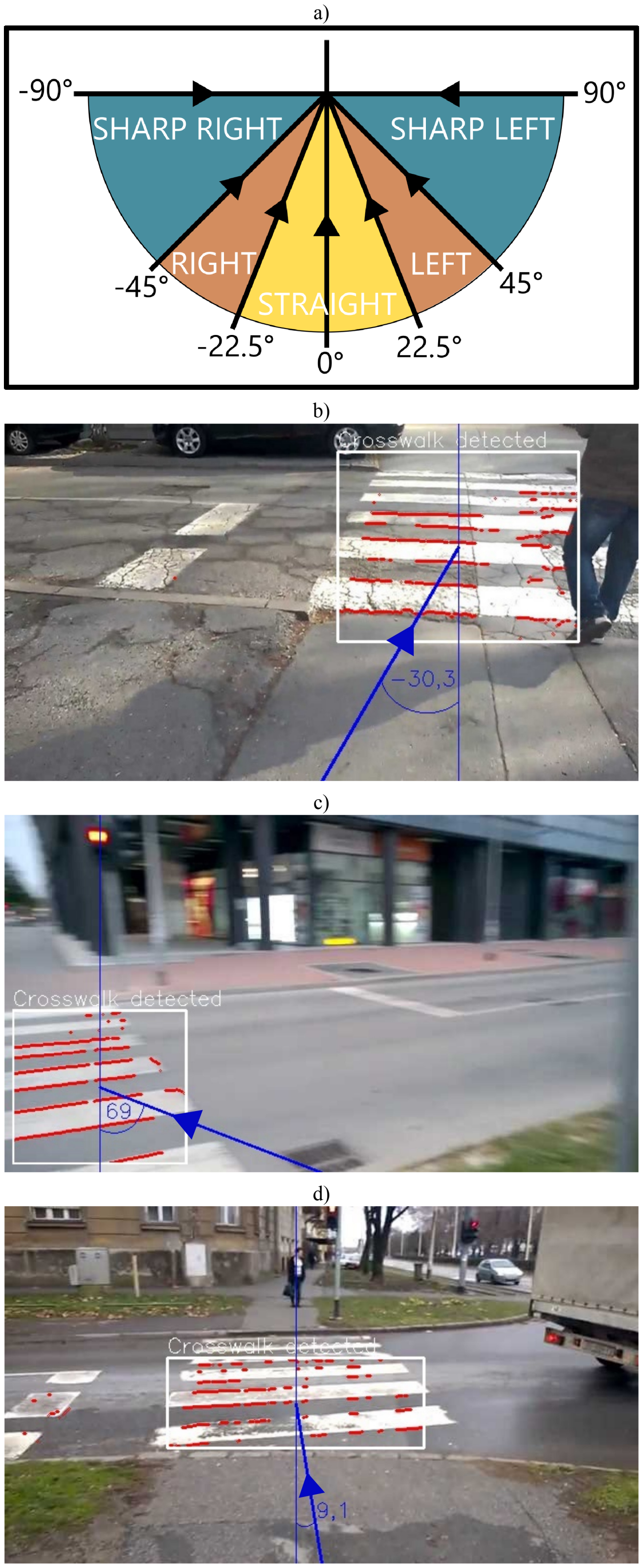


Figure 10. Determining the correct direction for the blind person: a) angle size mapped to 5 possible ways, b) for the angle of -30.3° proposed direction is RIGHT, c) for the angle of 69° proposed direction is SHARP LEFT, d) for the angle of 9.1° proposed direction is STRAIGHT

Angles are presented as positive and negative values relative to the vertical line which passes through center of the ROI. Examples of proposing directions based on angle sizes in particular cases are given in Fig. 10 b), c) and d).

IV. Pedestrian Walk Lights Recognition

This chapter explains the walk lights recognition process which is the next very important step when helping the blind in road crossing. Proposed algorithm for red or green walk light recognition starts with preprocessing whereupon 6-stage recognition is performed.

First fact important for light recognition is that it is not necessary to process the whole image while it is expected that the lights will be positioned somewhere above the crosswalk region as presented on Fig. 11. The whole following process is performed only on the region positioned above crosswalk region.

Figure 11. Walk light region above crosswalk region

Preprocessing part prepares original color image (Fig. 12 a) for detection of red or green light. For this purpose, red color is excessed and color image is converted to grayscale image where higher (whiter) values represent the extremely red areas (Fig. 12 b). This process is given by (4):

$$g(x,y) = f_R(x,y) - f_G(x,y) \tag{4}$$

where g is new gray value for coordinates (x,y), f_R is red channel value and f_G is green channel value. Such grayscale image is then thresholded with threshold value 35 to obtain binary image. All white shapes in this binary image (Fig. 12 c) are now potential candidate regions for the red light. Those regions are subjected to 6-stage recognition process.

Only in the case when red light is not detected, lights region is again processed to find potential green light. Green color is excessed by using following equation (5):

$$g(x,y) = f_G(x,y) - f_R(x,y) \tag{5}$$

Further process of thresholding is similar to one for the red lights. All white shapes on the binary image now represent the potential candidate regions for the green light.

All regions isolated in excessed red or green binary images are subjected to six stages of checks to determine whether one of the shapes is a walk light sign. Those stages are presented in Fig. 13 with specific conditions that need to be met for successful recognition. It is important to notice that walk light sign occupies small amount of pixels and the shape is not easily recognizable. The middle of the sign is often rather white than red or green due to bright illuminance captured by camera. Taking this into account, proposed recognition method is tailored to work with such low resolution images and realistic lighting conditions.

a)
b)
c)

Figure 12. Walk light region: a) before processing, b) grayscale with excessed red color, c) binary image with excessed red color

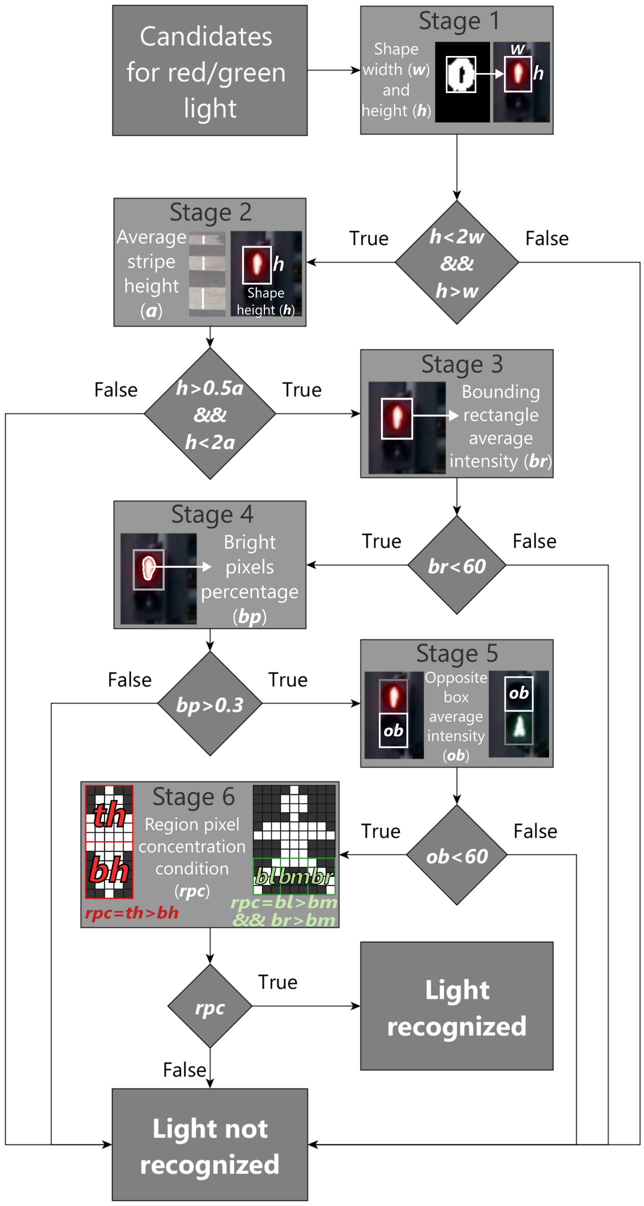


Figure 13. Six stages of walk light recognition with required conditions

In the **first stage**, bounding rectangle around the shape of the potential walk light is measured. Height (*h*) and width (*w*) of the bounding rectangle should be in specific aspect ratio. Height should be larger than width but smaller than double width. If this condition is satisfied, shape successfully passed the first stage.

If we observe the position of the pillar with walk lights, we can see that it is always positioned right next to the road (Fig. 10). If crosswalk stripe height is prescribed, it can be concluded that the height (*h*) of the walk light shape in image will be in a specific ratio with average crosswalk stripe height. Average stripe height (*a*) is calculated as average distance of two edge-points in the same column. It is experimentally obtained that walk light shape height should be at least a half of average stripe height. Shape height also has to be smaller than double of average stripe height. If those two conditions are satisfied, **the second stage** is passed.

The third stage is also in conjunction with bounding rectangle. The area around red or green walk sign is always in black or at least in dark colors so the pixels in that area should have small intensities. Average intensity of all pixels in bounding rectangular frame (*br*) is calculated and it should not exceed the value 60 to consider it for further recognition.

When capturing artificial light sources on camera, like those in walk lights, they are often bright and represented with nearly white color. Unlike the bounding frame in previous stage, middle of the selected rectangular area should include bright, almost white pixels. For **the fourth stage**, percentage of bright pixels (*bp*) must be more than 30%, where bright pixels are those with average intensity of three channels larger than 50.

Since the red walk light is always positioned above the green light, it is obvious that in cases when red light is turned on, opposite area bellow it will be mostly dark with dominating black colors. Similarly, when green light is turned on, opposite area above it will be mostly dark. **The fifth stage** checks the average pixel intensity of the opposite box (*ob*) which has to be smaller than 60 for the stage to be passed.

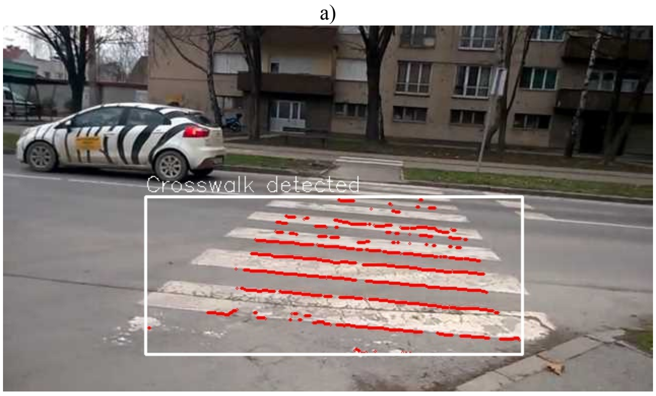
The last **sixth stage** is performed on binary image having white pixels on the positions where red or green channel intensity is higher than 150. For walk light region, this will give characteristic contours of standing or walking figure. Concentration of white pixels in some subregions of such images can be assumed for standing and walking figure, which allows us the recognition. Process is slightly different for red and green light. When trying to recognize red light, image is divided in two subregions where *bh* is the white pixel concentration in the bottom half and *th* in the top half. If there are more white pixels concentrated in top half, red light is recognized. For the recognition of green light, image is divided in equal ninths where only three bottom subregions are compared. Green light is recognized when concentration of white pixels in the bottom middle subregion (*bm*) is lower than in the bottom left (*bl*) and the bottom right subregion (*br*).

V. Results and Performance

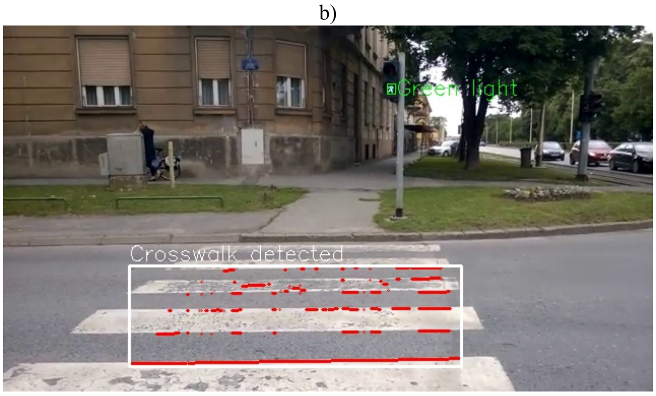
Obtained experimental results are presented in this chapter. The focus is on three main parameters: crosswalk detection accuracy, walk light recognition accuracy and processing speed.

For the purpose of testing the developed algorithm, video sequences were captured with camera mounted on person in the chest height to get first person view. All videos were captured on locations characteristic for walking routes. Videos include scenes when approaching the crosswalk but also the scenes without crosswalk. Proposed algorithm is tailored to detect crosswalks from the distance of 1 to 3 meters. Particular frames containing crosswalks were extracted from videos and dataset of 120 images is collected. All images are different and represent the variety of capturing conditions: sunny and cloudy weather, small and large sized crosswalks, partially ruined or concealed crosswalks. On the other side, another 120 images were extracted from scenes where crosswalk is not present. This dataset will be used to test the percentage of false positive detections. Detection examples are shown in Fig. 14. Examples include four images taken during the day light: with only crosswalk (Fig. 14 a), with crosswalk and green light (Fig. 14 b), with crosswalk and red light (Fig. 14 c) and without crosswalk (Fig. 14 d).

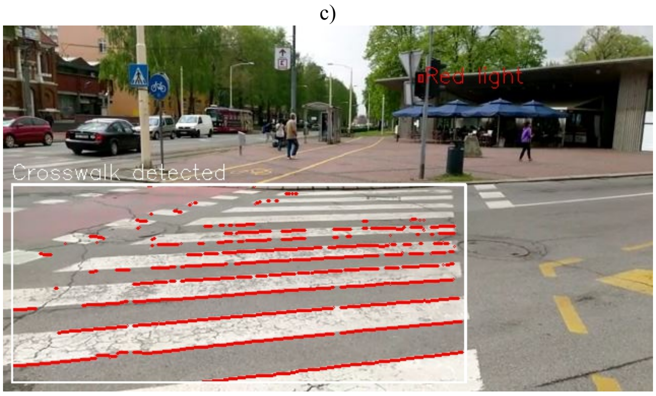
a)



b)



c)



d)

e)

f)

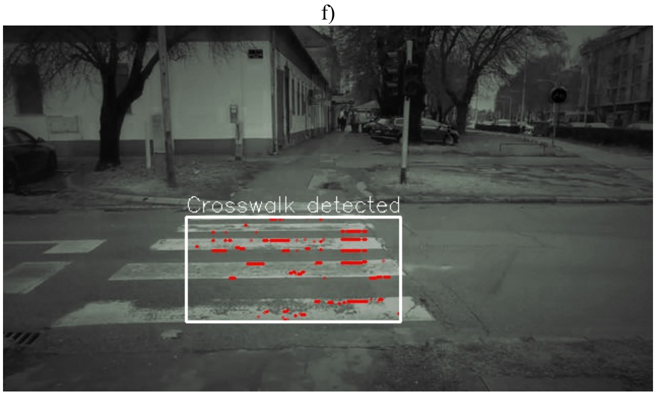


Figure 14. Detection examples: a) Crosswalk only, b) Crosswalk and green light, c) Crosswalk and red light, d) Without crosswalk, e) Crosswalk in low-contrast evening image, f) Crosswalk in simulated night scene with infrared light source

There are also two example images taken under aggravating lighting circumstances: low-contrast evening scene (Fig. 14 e) and simulated night scene with infrared light source (Fig. 14 f).

Proposed method gained very promising detection rates where 118 of 120 (98.33%) crosswalks were successfully detected. Two negative detections were caused by tree shadows over the crosswalk and vanished white stripes on aged crosswalk. On 120 images without crosswalks, there were only 3 false positive detections. When analyzing false positive detections, all 3 mistakes were caused by patterns on the ground similar to crosswalks. Compared to the similar methods, proposed method gained better results in the terms of detection accuracy. Several methods gained better results for false positive detection but those are tested on significantly smaller dataset. Method described in [5] has slightly different testing approach where video is taken on three specific routes and 13 of 15 crosswalks were detected, but false positives are tested on entire set of frames. Comparison of the results of similar methods is given in Table I.

TABLE I. ACCURACY COMPARISON OF METHODS FOR CROSSWALK DETECTION

Method	Correct detections			False positive detections		
	Dataset size	**Detected**	**%**	**Dataset size**	**Detected**	**%**
Proposed method	120	118	98.33	120	3	2.50
Segment-ation and projective invariant [7]	81	77	95.06	37	0	0
RGB-D [10]	52	41	78.85	70	0	0
Mean-shift segmenta-tion [6]	40	33	82.50	43	2	4.65
Template matching [5]	15	13	86.67	19903	30	0.15

Additional walk light recognition algorithm is tested on the same dataset as previous tests for crosswalk detection. Therefore, 57 of 120 images with crosswalk also contained walk light for pedestrians. There were 36 red lights and 21 green lights. Proposed 6-stage algorithm has yielded slightly better results for detection of red lights where 94.44% lights were successfully recognized. Green light was correctly recognized in 80.95% of cases, which brings overall percentage to 89.47%. Since the most of the similar methods deals with the traffic lights for vehicles, it is hard to compare obtained results with them. Authors in [10] performed the most similar experiment and compared to our proposed method their tests shown worse results for the red lights, but better results for the green lights. However, it is still hard to compare those methods because the one in [10] works for different kind of walk light sings characteristic for United States of America. On the other side, our algorithm is tailored to work on the walk light signs in European countries. Detailed results of our experiment on walk light recognition are shown in Table II.

Processing speed is crucial for algorithms that provide necessary information for the blind or visually impaired persons. In order to inform the user about potential crosswalk and walk light state on time, proposed algorithm is optimized and focused on avoiding the processing of unnecessary pixels. Proposed multiresolution approach allows us to process smaller amount of pixels thereby preserving the processing of regions with information crucial for detection. Table III shows average amount of pixels to process per image that can be reduced by using multiresolution approach. It is visible that two actions on low scale images can reduce amount of pixels to process for about 50.38% in first part of the method and then about 93.75% in second part of processing.

TABLE II. ACCURACY OF WALK LIGHT RECOGNITION

Red light	**Dataset size**	36
	Recognized	34
	%	94.44
Green light	**Dataset size**	21
	Recognized	17
	%	80.95
Overall	**Dataset size**	57
	Recognized	51
	%	89.47

TABLE III. COMPARISON OF THE AMOUNT OF PIXELS TO BE PROCESSED IN PARTICULAR STEPS

Action	**Pixels to process (average)**		**Reduction percentage**
	Full resolution	**Multi resolution**	
Preprocessing and edge-points detection	230 400	114325	**-50.38%**
Line-energy calculation and decision making	110720	6920	**-93.75%**

Except the number of pixels to process, processing speed depends on the technical specifications of the processing device. Since this algorithm is supposed to be used in movement, processing speed tests were conducted on portable battery-powered devices. The first device for speed test was a laptop with Intel i5-6200U processor and 8 GB of RAM. The second device was pocket-sized PC with Intel Atom Z3735F processor and 2 GB of RAM. Since it is highly portable device and has low power consumption, we found it as a suitable platform for the purpose of navigation for the blind. This device has similar hardware components as today's smartphones and tablets. Therefore, performance testing on this platform can help us to examine the possibility for further research and algorithm implementation on devices like smartphones and tablets.

Processing time is measured for every image in the dataset. Obtained values include time for both, detection of crosswalk and recognition of walk lights. Overall average processing speed values also include processing of images without crosswalks. As shown in Table IV, average time per image when processing on laptop is about 60 milliseconds, which means processing of more than 16 frames per second. This should ensure timely information about crosswalk and walk light to moving user. Processing speed on pocket-sized PC is lower as expected but still acceptable. With processing of 4 frames per second and average walking speed of 1.4 m/s this is still enough to provide information before standing on the crosswalk even if it is recognized from distance of only one meter. Obtained processing times are given in Table IV with detailed specifications of the processing devices.

TABLE IV. AVERAGE PROCESSING SPEED TESTED ON PORTABLE DEVICES

Device specifications	**Type**	Laptop	Pocket-PC
	Model	Lenovo V310 notebook	Lenovo Ideacentre Stick 300
	Processor	Intel i5-6200U @ 2.3 Ghz (2 cores)	Intel Atom Z3735F @ 1.33 GHz (4 cores)
	Memory	8 GB of RAM	2 GB of RAM
Average processing time (per image)		60.3 ms	218.1 ms
Frames per second		16.6	4.6

VI. CONCLUSION

After conducting experiments, it can be concluded that the computer vision based detection of crosswalks can be potentially applied in advanced digital aid for the blind and visually impaired. Proposed algorithm is very accurate when detecting crosswalks in various realistic situations. High detection rates and small amount of false positive detections shows robustness of the proposed method. Method is adjusted to work on video frames where crosswalk is partially obscured, faded or is situated on slightly textured road. Negative results appeared mostly because of unpredictable shapes of shadows on the ground and solving that problem is ongoing work.

Walk light recognition tests gained good results, especially when detecting red light. However, it is questionable whether it is possible the get better results using this input image resolution of 640×360, where walk light sometimes covers only few pixels. This leaves the space for further improvements of the recognition method or possible usage of higher resolutions for this purpose.

When talking about processing speed, proposed multiresolution approach significantly reduced the number of pixels that need to be processed in two main steps of the method. Processing speed also depends on processing device, which for this purpose must be portable in motion. When using mid-range laptop this method can provide information about crosswalk and walk light more than 16 times in a second, which certainly can be considered as real-time processing. With small pocket-sized PC, which is more compact and easier to carry in motion, average processing speed is 4.6 frames per second. Considering the standard walking speed, this should also be enough to provide information about crosswalk and walk light on time. According to the obtained results for accuracy and processing speed, it can be concluded that camera based systems will have a significant role in future assistive tools for the blind and visually impaired.

REFERENCES

[1] L. Hakobyan, J. Lumsden, D. O'Sullivan, and H. Bartlett, "Mobile assistive technologies for the visually impaired," Surv. Ophthalmol., vol. 58, no. 6, pp. 513–528, Nov. 2013. doi:10.1016/j.survophthal.2012.10.004

[2] D. Dakopoulos and N. G. Bourbakis, "Wearable Obstacle Avoidance Electronic Travel Aids for Blind: A Survey," IEEE Trans. Syst. Man

Cybern. Part C Appl. Rev., vol. 40, no. 1, pp. 25–35, Jan. 2010. doi:10.1109/TSMCC.2009.2021255

[3] R. Tapu, B. Mocanu, and E. Tapu, “A survey on wearable devices used to assist the visual impaired user navigation in outdoor environments,” in Electronics and Telecommunications (ISETC), 2014 11th International Symposium on, 2014, pp. 1–4. doi:10.1109/ISETC.2014.7010793

[4] J. Choi, B. T. Ahn, and I. S. Kweon, “Crosswalk and traffic light detection via integral framework,” in Frontiers of Computer Vision,(FCV), 2013 19th Korea-Japan Joint Workshop on, 2013, pp. 309–312. doi:10.1109/FCV.2013.6485511

[5] 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, 2014, pp. 285–290. doi:10.1109/MECATRONICS.2014.7018597

[6] M. Radvanyi, B. Varga, and K. Karacs, “Advanced crosswalk detection for the Bionic Eyeglass,” in 12th International Workshop on Cellular Nanoscale Networks and Their Applications (CNNA), 2010, pp. 1–5. doi:10.1109/CNNA.2010.5430281

[7] M. S. Uddin and T. Shioyama, “Detection of Pedestrian Crossing Using Bipolarity Feature-an Image-based Technique,” Trans Intell Transp. Sys, vol. 6, no. 4, pp. 439–445, Dec. 2005. doi:10.1109/TITS.2005.858787

[8] J. M. Coughlan and H. Shen, “A fast algorithm for finding crosswalks using figure-ground segmentation,” in Proc. 2nd Workshop on Applications of Computer Vision, in conjunction with ECCV, 2006.

[9] J. M. Coughlan and H. Shen, “Crosswatch: a System for Providing Guidance to Visually Impaired Travelers at Traffic Intersections,” J. Assist. Technol., vol. 7, no. 2, Apr. 2013. doi:10.1108/17549451311328808

[10] S. Wang, H. Pan, C. Zhang, and Y. Tian, “RGB-D image-based detection of stairs, pedestrian crosswalks and traffic signs,” J. Vis. Commun. Image Represent., vol. 25, no. 2, pp. 263–272, Feb. 2014. doi:10.1016/j.jvcir.2013.11.005

[11] M. Poggi, L. Nanni, and S. Mattoccia, “Crosswalk Recognition Through Point-Cloud Processing and Deep-Learning Suited to a Wearable Mobility Aid for the Visually Impaired,” in New Trends in Image Analysis and Processing -- ICIAP 2015 Workshops, 2015, pp. 282–289. doi:10.1007/978-3-319-23222-5_35

[12] Y. Jie, C. Xiaomin, G. Pengfei, and X. Zhonglong, “A new traffic light detection and recognition algorithm for electronic travel aid,” in Intelligent Control and Information Processing (ICICIP), 2013 Fourth International Conference on, 2013, pp. 644–648. doi:10.1109/ICICIP.2013.6568153

[13] V. Ivanchenko, J. Coughlan, and H. Shen, “Real-time walk light detection with a mobile phone,” in Computers Helping People with Special Needs, Springer, 2010, pp. 229–234. doi:10.1007/978-3-642-14100-3_34

[14] S. Mascetti, D. Ahmetovic, A. Gerino, C. Bernareggi, M. Busso, and A. Rizzi, “Robust traffic lights detection on mobile devices for pedestrians with visual impairment,” Comput. Vis. Image Underst., vol. 148, pp. 123–135, Jul. 2016. doi:10.1016/j.cviu.2015.11.017

[15] K. Romić, I. Galić, and H. Leventić, “Influence of the input image resolution on the staircase detection,” in ELMAR, 2016 International Symposium, 2016, pp. 177–180. doi:10.1109/ELMAR.2016.7731781

[16] W. Hackbusch, “Multi-Grid Methods and Applications”, pp. 80-96, Springer Science & Business Media, 2013.

[17] C. Fernandez-Maloigne, F. Robert-Inacio, and L. Macaire, “Digital Color: Acquisition, Perception, Coding and Rendering”, pp. 65-90, John Wiley & Sons, 2013.

[18] W. Burger and M. J. Burge, “Digital Image Processing: An Algorithmic Introduction Using Java”, pp. 291-325, Springer, 2016.

[19] P. Soille, “Morphological Image Analysis: Principles and Applications”, pp. 105-135, Springer Science & Business Media, 2013.

[20] K. Romić, I. Galić, and T. Galba, “Technology assisting the blind—Video processing based staircase detection,” in ELMAR (ELMAR), 2015 57th International Symposium, 2015, pp. 221–224 doi:10.1109/ELMAR.2015.7334533