Journal of Electronic Imaging

JElectronicImaging.org

Crosswalk navigation for people with visual impairments on a wearable device

Ruiqi Cheng
Kaiwei Wang
Kailun Yang
Ningbo Long
Weijian Hu
Hao Chen
Jian Bai
Dong Liu

Ruiqi Cheng, Kaiwei Wang, Kailun Yang, Ningbo Long, Weijian Hu, Hao Chen, Jian Bai, Dong Liu, "Crosswalk navigation for people with visual impairments on a wearable device," *J. Electron. Imaging* **26**(5), 053025 (2017), doi: 10.1117/1.JEI.26.5.053025.

Crosswalk navigation for people with visual impairments on a wearable device

Ruiqi Cheng, Kaiwei Wang,* Kailun Yang, Ningbo Long, Weijian Hu, Hao Chen, Jian Bai, and Dong Liu
Zhejiang University, College of Optical Science and Engineering, Hangzhou, China

Abstract. Detecting and reminding of crosswalks at urban intersections is one of the most important demands for people with visual impairments. A real-time crosswalk detection algorithm, adaptive extraction and consistency analysis (AECA), is proposed. Compared with existing algorithms, which detect crosswalks in ideal scenarios, the AECA algorithm performs better in challenging scenarios, such as crosswalks at far distances, low-contrast crosswalks, pedestrian occlusion, various illuminances, and the limited resources of portable PCs. Bright stripes of crosswalks are extracted by adaptive thresholding, and are gathered to form crosswalks by consistency analysis. On the testing dataset, the proposed algorithm achieves a precision of 84.6% and a recall of 60.1%, which are higher than the bipolarity-based algorithm. The position and orientation of crosswalks are conveyed to users by voice prompts so as to align themselves with crosswalks and walk along crosswalks. The field tests carried out in various practical scenarios prove the effectiveness and reliability of the proposed navigation approach. *© 2017 SPIE and IS&T* [DOI: 10.1117/1.JEI.26.5.053025]

Keywords: adaptive extraction and consistency analysis algorithm; stripe clustering; crosswalk navigation; blind assistance.
Paper 170532 received Jul. 9, 2017; accepted for publication Oct. 3, 2017; published online Oct. 27, 2017.

1 Introduction

It is inconvenient for people with visual impairments to travel outdoors due to the lack of capability to perceive surrounding obstacles and hazards. To assist people with visual impairments to avoid obstacles, various techniques of traversable area detection in a wearable system have been achieved in our previous work.[1–3] However, the crosswalk detection was not implemented in the wearable system. Crosswalk perception at urban intersections is one of the most important demands for people with visual impairments. Zebra crosswalks are commonly used in many countries, such as China, Italy, and France. In this paper, a crosswalk detection algorithm and its interactive approach are elaborated to provide people with visual impairments with zebra crosswalk navigation when crossing roads.

With the development of computer vision and smart devices, many researchers are dedicated to helping people with visual impairments to cross roads.[4–16] Most of the detection algorithms can be categorized into three types of methodologies: edge-based algorithms, gray-scale pattern-based algorithms, and bright stripe-based algorithms.

1.1 *Edge-Based Algorithms*

A crosswalk is composed of several bright (white or yellow) stripes and dark background, so parallel straight edges of crosswalk stripes are a kind of feature of crosswalks. Hödlmoser et al.[8] detected crosswalks by acquiring the long edges of the stripes. The edges are extracted by a Canny edge detector and then merged by a random sample consensus (RANSAC) algorithm. Similarly, Wei et al.[12] applied edge detector and Hough transformation to crosswalk detection. Mascetti et al.[9,17] used an inertial measurement unit for ground plane reconstruction, and they utilized edge-based detection and consistency checks to extract stripes and structured them into crosswalks. The algorithms perform well on straight-line crosswalks, whereas it is not valid on polyline crosswalks. As shown in Fig. 1, polyline crosswalks, which are composed of polyline stripes, instruct the walking direction for pedestrians.

1.2 *Gray-Scale Pattern-Based Algorithms*

As shown in Fig. 1, bright stripes alternate with a dark background in crosswalks, so the periodic gray value distribution is also a feature of crosswalks. An assistant device for people with visual impairments based on template matching of a binary image was elaborated in Ref. 10, and it requires the camera to be orthogonal to high-contrast crosswalks. Considering the sharp contrast near the boundary of black–white stripes, Uddin and Shioyama[4] detected crosswalks by analyzing bipolarity of a gray-scale histogram. Nevertheless, the performance of the algorithm is sensitive to the predetermined segmenting size, which has to be switched for different scenarios. Poggi et al.[14] proposed a crosswalk recognition algorithm that is based on plane detection and a convolutional neural network, but it needs a RGB-D camera.

1.3 *Bright Stripe-Based Algorithms*

In these algorithms, the bright stripes of crosswalks are extracted and analyzed. Zhai et al.[15] developed a crosswalk detection algorithm based on maximally stable extremal regions and extended RANSAC. Although the algorithm performs well under different illuminations, it is more suitable for traffic surveillance images, where crosswalk stripes are identical in shape and the background can easily be eliminated. Coughlan and Shen[5] used figure-ground segmentation to detect crosswalks; they established the relationship

*Address all correspondence to: Kaiwei Wang, E-mail: wangkaiwei@zju.edu.cn

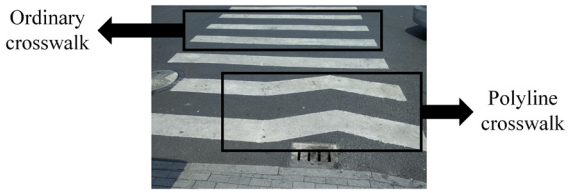

Fig. 1 Polyline crosswalks.

between the width and the position of stripes in images. The algorithm detects crosswalks well under different scenarios, but the efficiency is not high since a few seconds are needed to process one image.

Many detection algorithms[4,7,10,12,14,18,19] focus on simple scenarios, where crosswalks are usually in front of users and take a large proportion of a whole image. However, in practical crosswalk navigation, crosswalks may present different orientations and may locate at any position within the field of view. For those algorithms, the crosswalks at a far distance from users cannot be detected, which restricts the range of crosswalk navigation. We aim to tackle the problem in this paper.

A typical scenario of crosswalk navigation is presented in Fig. 2. The green point denotes the position of the user, and the yellow line denotes the horizontal field of view of the camera. Our system aims to detect the crosswalks both in front of [Fig. 2(c)] and at a distance from users [Figs. 2(a) and 2(b)]. Meanwhile, voice prompts are recorded to instruct the user to align with the crosswalk.

As an assistance approach for people, the detection algorithm should deliver low-false alarms, which is of vital importance for the safety of users;[17] real time is another requirement for the algorithm to maintain a moderate frame rate, because the algorithm is implemented in a portable platform with limited resources;[17] furthermore, the robustness must be ensured for the compatibility with various scenarios, such as various illuminations, low-contrast crosswalks, and pedestrian occlusion.[17]

In this paper, a crosswalk detection algorithm and interactive approach are presented. The rest of this paper is organized as follows: in Sec. 2, we present the adaptive extraction and consistency analysis (AECA) algorithm, which is used to detect crosswalks from images. Section 3 shows the optimization and performance of the detection algorithm. The interactive approach and field tests are described in Sec. 4. A brief conclusion is drawn in Sec. 5.

2 Real-Time Crosswalk Detection Algorithm

To detect crosswalks, we propose the AECA algorithm, which is shown in Fig. 3. First, adaptive thresholding, where the threshold is determined by the neural network, extracts bright-connected components (also called candidates) from a gray-scale image. A connected component, which is a prospective crosswalk stripe, is defined as a subset of images where any two points of the subset are connected.[20] The extraneous candidates, such as the sky, are then removed by analyzing the geometrical properties of candidates. Meanwhile, the candidates, which are similar in shape, are merged into a new candidate, so as to merge crosswalk stripes, which are split by obstacles. Finally, those candidates with consistent features are selected by consistency analysis to form the crosswalk. The position and orientation of the crosswalk are also obtained.

2.1 *Adaptive Candidate Extraction*

In contrast to the dark background, the stripes of crosswalks are bright; thus, they are possible to be extracted from images. Intended to extract the bright stripes of crosswalks, we utilize a neural network-based binary thresholding approach to adapt to different environmental illuminations. To reduce the computing complexity and improve the performance of consistency analysis, the candidates are pruned by analyzing their geometrical properties. Occasionally, pedestrians, as well as obstacles, may occlude crosswalks, thus split candidates need to be merged into an intact candidate to achieve complete crosswalk detection.[9,17]

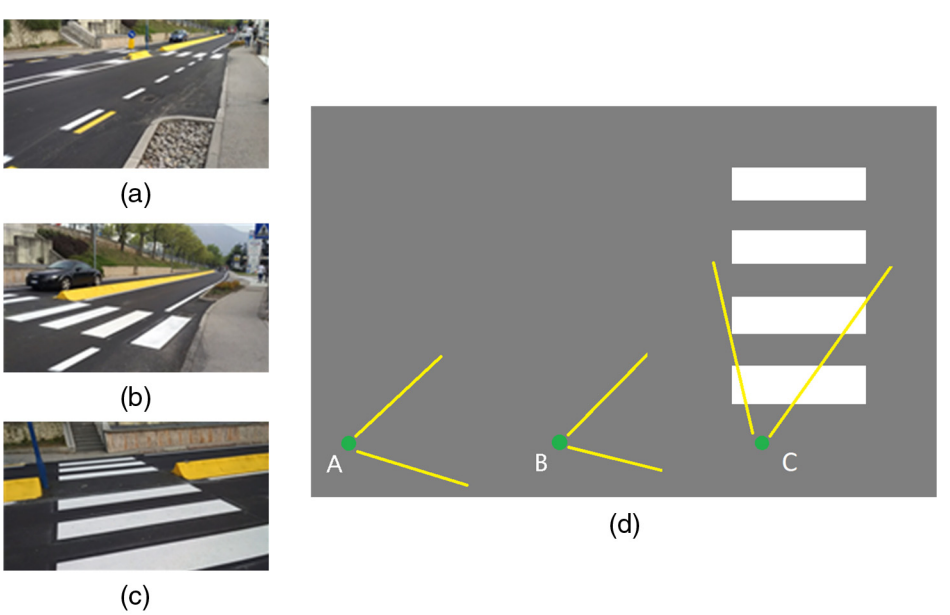

Fig. 2 A typical scenario of crosswalk detection, (a)–(c) correspond to the images captured at A–C in (d).

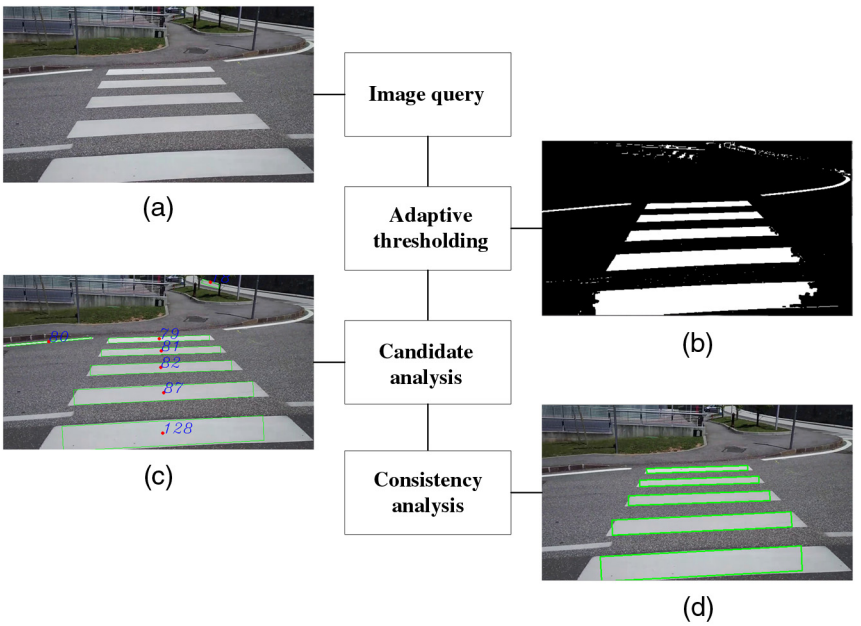

Fig. 3 The flowchart of AECA algorithm. The processing procedures include adaptive candidate extraction (adaptive thresholding and candidate analysis) and stripe consistency analysis (clustering). The lateral images (a)–(d) present the results of corresponding procedures.

2.1.1 *Adaptive thresholding based on neural network*

Acquired color frames from the camera are transformed into gray-scale images, since the brightness of crosswalks is used in binary thresholding and the hue of crosswalks could be neglected. To segment images properly, the threshold should lie between the gray value of bright stripes and dark background. The fixed thresholding algorithm has a poor performance, because the threshold is predetermined and does not vary with diverse illuminations. Herein, the adaptive thresholding is adopted to address this issue, where the threshold is not fixed but determined by the input image.

An artificial neural network is a commonly used model to deal with the adaptive binary thresholding problems. In this paper, a three-layer feed-forward neural network is used to express the relation between a gray-scale image and binarization threshold. As shown in Fig. 4, the neural network is comprised of the input layer, the hidden layer, and the output layer. Each neuron in the hidden layer is directly linked with the neurons in input and output layers. The input layer $\mathbf{X} = (x_1, x_2, \ldots, x_i)$ receives the down-sampling gray-scale image (tiny image). The original gray-scale image is resampled to a tiny image and unfolded to form the vector $\mathbf{X}$. Each neuron in the output layer $\mathbf{Y} = (y_1, y_2, \ldots, y_k)$ denotes the possibility that the corresponding gray value in $\mathbf{G} = (g_1, g_2, \ldots, g_k)^{\mathrm{T}}$ is the suitable threshold of input image $\mathbf{X}$. Therefore, adaptive thresholding is converted into a classification task by the neural network, and vector $\mathbf{G} = (g_1, g_2, \ldots, g_k)^{\mathrm{T}}$ contains the values of the binarization threshold (namely classes). Thereby, the suitable threshold t is determined by the product of possibilities and threshold values

$$t = \mathbf{YG}. \tag{1}$$

Having determined the threshold t, we carry out binary thresholding on the original gray-scale image and get the binary image [see Figs. 5(a) and 5(d)]. The bright pixels are grouped to form connected components, also called candidates in this paper.

2.1.2 *Candidate analysis*

The geometrical properties of candidates are analyzed to remove the undesired candidates generated by adaptive thresholding. The size, orientation, and convexity of each

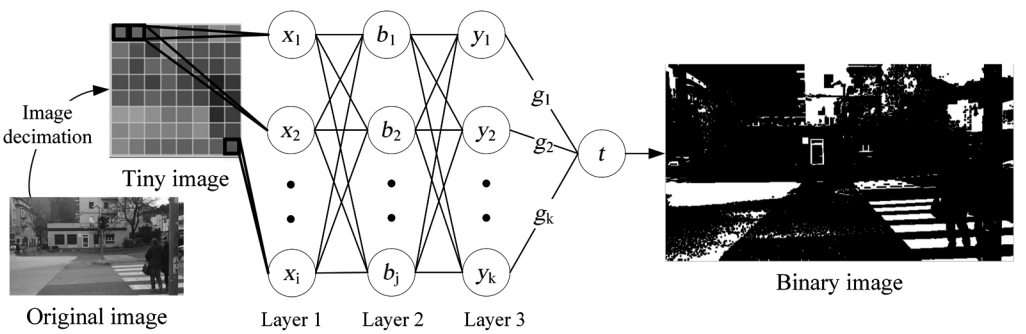

Fig. 4 The structure of neural network for adaptive thresholding.

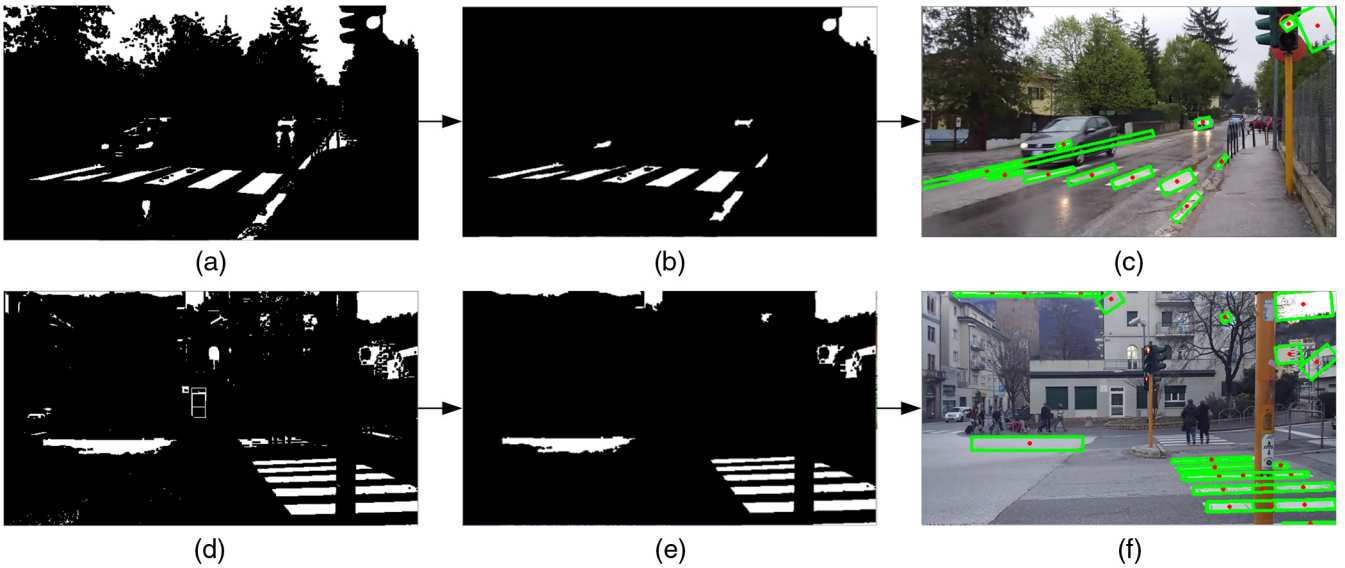

Fig. 5 The results of candidate analysis. (a) and (d) Adaptive binary thresholding. (b) and (e) Candidate pruning by size, orientation, and convexity. (c) and (f) Candidate merging, where candidates are presented as green boxes with red centroids.

candidate are examined, and only those candidates that conform to all of the criteria are retained. In addition, candidates with similar geometrical properties (e.g., orientation, width, and position in the image) are merged, which restores the crosswalk stripes separated by pedestrians or other objects.

The size filter is designed to eliminate enormous and tiny candidates, such as bright pebbles and sky. It does not aim to select crosswalk stripes from candidates exactly. Therefore, the upper bound and lower bound of the size filter could be determined empirically. The maximum and minimum sizes of candidates (in pixel) are defined as

$$\text{size}_{\max} = 0.2\ HW \quad \text{and} \quad \text{size}_{\min} = 5 \times 10^{-4}\ HW,$$

where H and W denote the height and width of the acquired frame. To avoid eliminating prospective crosswalk stripes, the interval of the upper bound and the lower bound is intentionally enlarged. For each candidate, the fitting ellipse is calculated using the algorithm in Ref. 21. The length (the length of major axis), width (the length of minor axis), and orientation (the angle of major axis) of the fitting ellipse are defined as the length, width, and orientation of the corresponding candidate, respectively.

The crosswalk stripes are vertical in images, when the optical axis of the camera is parallel to the long edge of stripes and the user is in the middle of the road or far from crosswalks. However, in our scenarios of crosswalk navigation (Fig. 2), that case is not considered. Hence, the candidate which has a small angle (e.g., <20 deg) with the vertical direction is removed. The crosswalk stripe is convex, which means that its centroid is within its boundary. Thereby, the candidate whose centroid is out of the boundary is removed. The processing results after candidate reduction are shown in Figs. 5(b) and 5(e).

Two or more candidates with identical orientation and width (child candidates) are merged into one new candidate (parent candidate), if the orientations of child candidates are identical to the link of their centroids. Later, the parent candidate and its child candidates simultaneously exist among candidates, as shown in Figs. 5(c) and 5(f).

All of the remaining candidates are collected to form candidate set C, which will be dealt with in consistency analysis.

2.2 *Consistency Analysis*

The candidate sets are defined as $C = \{s_k | 1 \le k \le n\}$, where n is the size of the set, and s_k is the candidate with the index k. In set C, crosswalk stripes and extraneous candidates simultaneously exist. Fortunately, the crosswalk stripes possess consistency, such as consistent color, consistent orientation, etc.; hence, it is possible to select them from the candidate set by consistency analysis.

Consistency analysis, which is derived from the RANSAC algorithm,[22] selects the optimal consistency set S^* from candidate set C as shown in Fig. 6. Consistency analysis obtains different consistency sets by starting with different initial candidate combinations. In the proposed consistency analysis, both the model constructed by initial candidates and the clustering rules to gather consistent candidates are specially designed for crosswalk detection.

Two different candidates (s_i, s_j) are chosen as the initial candidates from set C. In view of the limited size of set C, all of the pairwise initial candidates (parent–child pairs excluded) will be traversed in the algorithm. Consistency set S_{ij} is defined as the clustering result starting with candidates s_i and s_j. To set a baseline for candidate clustering, a straight line l_{ij} is determined by the centroids of s_i and s_j. The line simulates the extending direction of the crosswalk. Starting with s_i and s_j, the consistent candidate s_k is added into S_{ij}, if it conforms with consistency criteria as follows:

1. s_k must be close to line l_{ij}, because the stripes of crosswalks are bound to gather along the extending direction.[15] Ideally, the centroid of candidate s_k should be on the line l_{ij}. The error tolerance of distance between candidate centroid and line is defined as E_l.

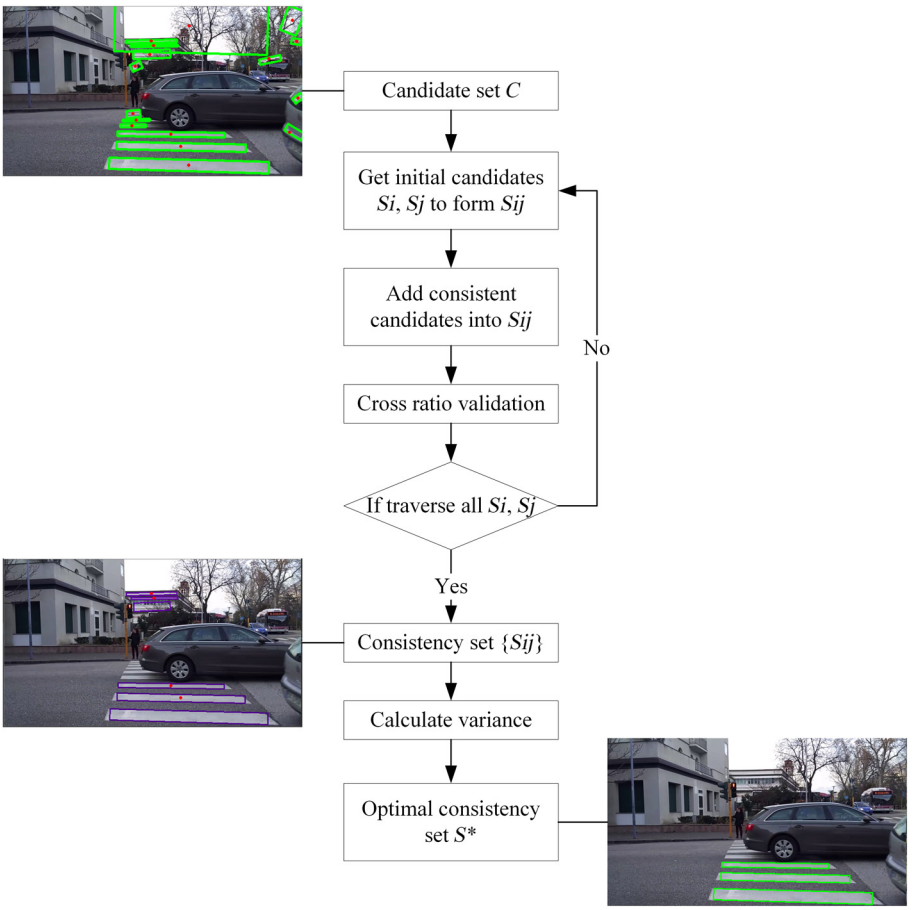

Fig. 6 The flowchart of consistency analysis.

2. The orientation of s_k is not in parallel with line l_{ij}, because the extending direction of crosswalks is orthogonal to the orientation of bright stripes. Here, we define the lower bound of angle difference between line and candidate as E_a.
3. Taking s_k as a quadrilateral, the opposite edges of s_k must intersect with line l_{ij}. Notice that the adjacent edges of s_k should not intersect with line l_{ij}.

Parent and child candidates are mutually exclusive among consistency set S_{ij}. If they exist simultaneously, child candidates are pruned. To make candidates in set S_{ij} more consistent, the color and orientation of candidates are also considered. The mean color and orientation are calculated among all the candidates in S_{ij}. The candidate, whose color or orientation is far from the mean value, is removed from set S_{ij}. The tolerances of candidate color and orientation are defined as E_c and E_o, respectively. After removing unqualified candidates from set S_{ij}, line l_{ij} is again fitted using all the candidates in S_{ij} by least squares method.

The cross ratio, defined in Ref. 4, is an important perspective-invariant property for crosswalks. Therefore, validating the cross ratio of candidates in S_{ij} is an important approach to eliminate outliers. Along the fitting line l_{ij}, all of the candidates in S_{ij} are sorted by their position. Four points are located at the intersection of fitting line l_{ij} and edges of two adjacent candidates in set S_{ij}. The cross ratio, derived from the four points, should be equal to 0.25, if the width of crosswalk stripe is equal to the width of the dark interval. The practical cross ratio may vary with a different stripe–interval ratio. Thereby, the lower bound and upper bound for the cross ratio are defined as E_{rmin} and E_{rmax}. From the beginning to the ending of set S_{ij}, each two adjacent candidates (e.g., s_p, s_q, $p < q$) are utilized to calculate the cross ratio. If the cross ratio falls into $[E_{\text{rmin}}, E_{\text{rmax}}]$, s_p and s_q are regarded as two adjacent stripes of a crosswalk. On the contrary, s_p and s_q do not belong to the same crosswalk, and S_{ij} is segmented into two subsets with s_p as the ending of the first subset and s_q as the beginning of the second subset.

After examining entire cross ratios, only the candidates of the largest subset are retained in set S_{ij}. Generally, the number of crosswalk stripes is larger than 3. Thereby, we define the minimum size of the consistency set as 3, and the consistency set whose size is smaller than the minimum size is abandoned.

For the final consistency sets $\{S_{ij}\}$, we calculate variance of each S_{ij} as Variance $= \mathbf{D}^{\mathrm{T}}\mathbf{C}$. Each element in vector $\mathbf{D} = [D_1\, D_2\, D_3\, D_4]$ is a variance that describes a sort of dispersion degree of S_{ij}. Vector $\mathbf{C} = [C_1\, C_2\, C_3\, C_4]$ denotes the weights for corresponding variances. The detailed definitions of variances are included in Table 1, where N is the number of candidates in consistency set S_{ij}.

In the variance of length and width, different estimated values (L_r' and W_r') for each candidate, instead of the same mean value, are utilized to measure the dispersion of the set, because the length and width of stripes have intrinsic

Table 1 Variances definition and significance.

Definition of variance	Additional definition	Significance
$\mathbf{D}_1 = \frac{1}{N}\sum_{0<r<N} d_r^2$	—	Variance between candidates' centroids to the fitting line l
$\mathbf{D}_2 = \frac{1}{N}\sum_{0<r<N}(L_r - L_r')^2$	$L_r' = ay_r + b$	Variance of length among candidates
$\mathbf{D}_3 = \frac{1}{N}\sum_{0<r<N}(W_r - W_r')^2$	$W_r' = cy_r + d$	Variance of width among candidates
$\mathbf{D}_4 = \frac{1}{N}\sum_{0<r<N}(O_r - o_m)^2$	$o_m = \frac{1}{N}\sum_{0<i<N} O_i$	Variance of orientation among candidates

dispersion due to perspective. The perspective relation is expressed by a linear model, and the parameters (a, b, c, and d) are fitted by the least square method using candidates in consistency set S_{ij}.

Among $\{S_{ij}\}$, the set with the smallest variance (S') is regarded as the most consistent set. However, it is not necessarily the optimal set, in view that the small set is prone to have small variance. To make the detected crosswalk as intact as possible, the optimal consistency set S^* is defined as the largest set, which contains all of the candidates in S'.

The optimal consistency set S^* is the desired crosswalk detection result, and then we assume that the crosswalk is composed of candidates in S^* and the mean orientation of candidates in S^* (o_m) is the orientation of the crosswalk. If none of the consistency set exists, we believe that the crosswalk does not exist in the current image.

3 Parameter Optimization and Detection Performance

3.1 *Optimization of Parameters in Adaptive Thresholding*

The adaptive thresholding training dataset $\{(\mathbf{X}_l, t_l)|1 \le l \le 277\}$ (available at Ref. 23) is taken advantage of to determine the optimal structure and parameters of the neural network. Herein, $\mathbf{X}_l$ is a down-sampled image that contains crosswalks, t_l is the manually labeled threshold that separates crosswalks from background in $\mathbf{X}_l$, and 277 is the number of training data. According to Eq. (1), we construct the output layer $\mathbf{Y}_l = (y_1, y_2, \ldots, y_k)$ of the training data $(\mathbf{X}_l, t_l)$ by manually labeled threshold t_l and vector $\mathbf{G} = (g_1, g_2, \ldots, g_k)^{\mathrm{T}}$

$$\mathbf{Y}_l = \arg\min_{\mathbf{Y}} |t_l - \mathbf{YG}| \quad \text{s.t. } \mathbf{Y}_l \in \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_k\}, \tag{2}$$

where $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_k\}$ are the natural bases of $\mathbf{R}^k$ (k-dimensional vector space), namely $\mathbf{Y}_l$ has only one nonzero component and it equals 1.

Apparently, the vector $\mathbf{G}$ is required to construct the output layer for training data. The number of components in $\mathbf{G}$ is identical to that of the neurons in output layer $\mathbf{Y}$. The values of components in $\mathbf{G}$ influence the number distribution of training data among different classes (different neurons of output layers). Therefore, a suitable $\mathbf{G}$, which results in balanced data between classes, is important for the performance of adaptive thresholding. In Table 2, the number k and values of components in $\mathbf{G}$ are presented, and the numbers of training data in different classes are also listed.

For high generalization performance, an adaptive thresholding testing dataset (available at Ref. 23) is utilized to optimize the structure of the neural network. In testing data, the position of a crosswalk in an image and the suitable binary threshold are labeled. Hence, for the crosswalk region (cr) in an image, the error (D_a) between adaptive thresholding (I_a) and thresholding with the labeled threshold (I_l) is taken as the criterion to evaluate the performance of neural structure, which is

$$D_a = \sum_{(i,j)\in \mathrm{cr}} |I_a(i,j) - I_l(i,j)|. \tag{3}$$

Obviously, the better the neural network performs, the less error D_a has. Using training data, 80 different network structures are trained by the resilient propagation algorithm.[24] Among those structures, the neural network with the 9×9 tiny image (namely 81 neurons in input layer) and 24 neurons in hidden a layer achieves the least error D_a. Therefore, the network with 81 neurons in the input layer ($i = 81$) and

Table 2 The training data distribution among the different classes of neural network.

G	Y	Number of data
$g_1 = 30$	$\mathbf{e}_1$	6
$g_2 = 80$	$\mathbf{e}_2$	21
$g_3 = 110$	$\mathbf{e}_3$	20
$g_4 = 130$	$\mathbf{e}_4$	25
$g_5 = 140$	$\mathbf{e}_5$	39
$g_6 = 150$	$\mathbf{e}_6$	21
$g_7 = 160$	$\mathbf{e}_7$	37
$g_8 = 170$	$\mathbf{e}_8$	48
$g_9 = 180$	$\mathbf{e}_9$	32
$g_{10} = 195$	$\mathbf{e}_{10}$	21
$g_{11} = 215$	$\mathbf{e}_{11}$	7
Total		277

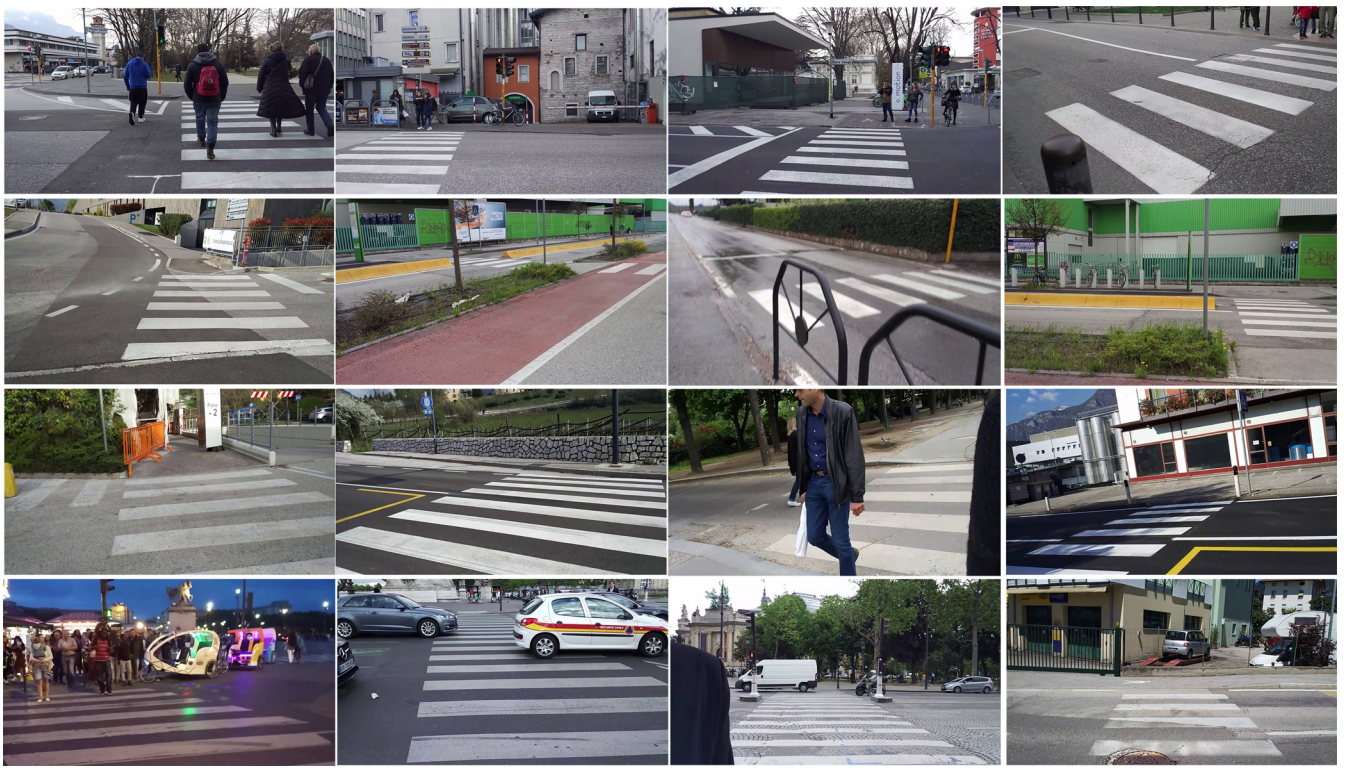

Fig. 7 Data set for tuning parameters in the algorithm.

24 neurons in the hidden layer ($j = 24$) is chosen as the optimized neural network.

To prove that the network-based thresholding is eligible to extract crosswalks from the background, Otsu's method,[25] as a reference, is also applied to the adaptive thresholding testing dataset. The difference (D_o) between Otsu's thresholding (I_o) and thresholding with the labeled threshold (I_l) is computed as

$$D_o = \sum_{(i,j)\in \mathrm{cr}} |I_o(i,j) - I_l(i,j)|. \tag{4}$$

For each testing datum, we get a ratio r of optimized neural network error D_a and Otsu's method error D_o as

$$r = \frac{D_a}{D_o}. \tag{5}$$

The mean ratio among testing datasets is 0.86, which demonstrates that the proposed adaptive threshold is superior to Otsu's method.

Table 3 Parameter combination of detection algorithm.

Parameter	Initial value	Minimal value	Maximal value	Optimized value
E_l	0.03 W	0.001 W	0.2 W	0.1 W
E_a	20	0	30	15
E_c	100	25	200	50
E_o	5	0	20	10
E_{rmin}	0.1	0.05	0.25	0.2
E_{rmax}	1	0.25	2	0.6
C_1	0.5	0	1	0.05
C_2	0.5	0	1	0.3
C_3	0.5	0	1	0.3
C_4	0.5	0	1	1

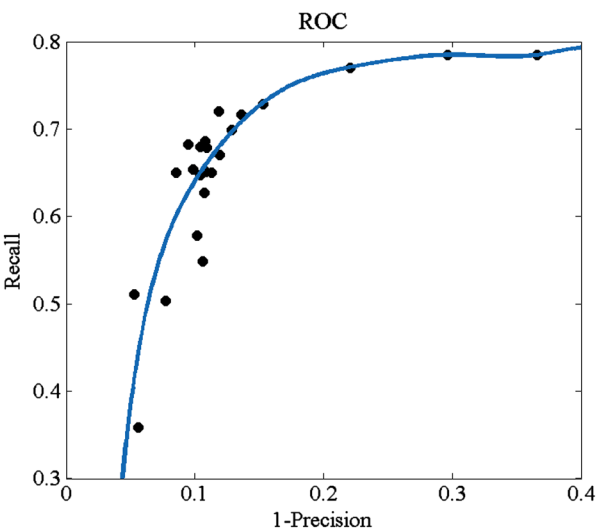

Fig. 8 ROC curve on training dataset.

Fig. 9 Crosswalk detection results using AECA algorithm on testing dataset. The detected stripes of crosswalks are labeled with green borders. (a) The detection results of crosswalks at far distances, (b) the detection results of crosswalks at close distances, (c) the detection results of crosswalks in front of users, (d) the crosswalks detection results in the scenarios of low contrast, and (e) the detection results where crosswalks are partially occluded.

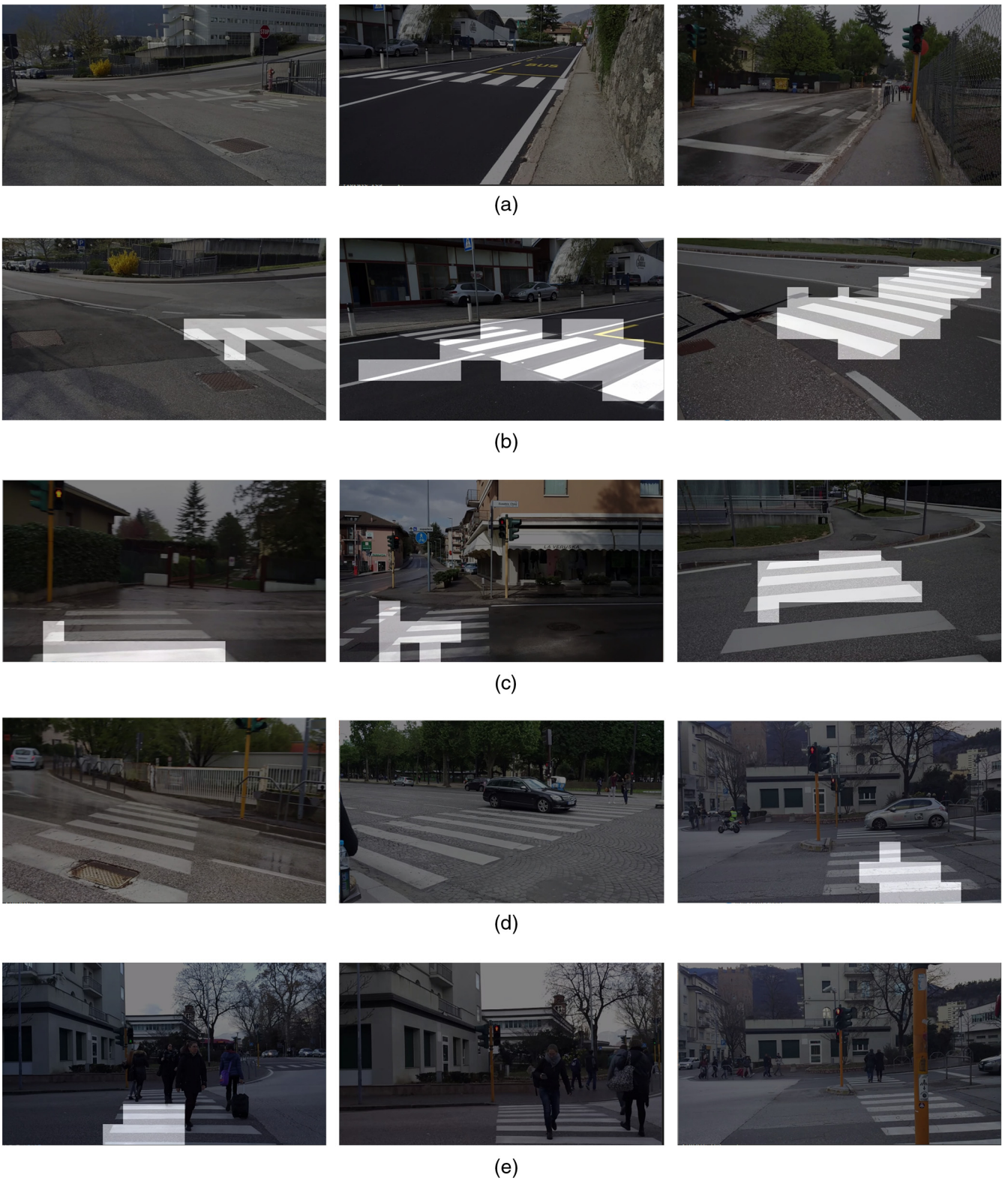

Fig. 10 Crosswalk detection results using bipolar algorithm on testing dataset. The detected crosswalk is labeled with white mask. (a) The detection results of crosswalks at far distances, (b) the detection results of crosswalks at close distances, (c) the detection results of crosswalks in front of users, (d) the crosswalks detection results in the scenarios of low contrast, and (e) the detection results where crosswalks are partially occluded.

Table 4 Recall and precision of crosswalk detection.

Algorithm	Scenarios	FN	FP	TN	TP	Recall (%)	Precision (%)
AECA	Total	122	39	104	187	60.1	84.6
	Occlusion	16	7	0	20	55.6	74.1
	Far	35	2	2	26	42.6	92.9
	Close	9	1	0	26	74.3	96.3
	Frontal	76	24	9	116	60.4	82.9
Bipolarity	Total	159	36	95	162	50.5	81.8
	Occlusion	25	0	3	15	37.5	100.0
	Far	49	10	2	4	7.5	28.6
	Close	11	1	0	24	68.6	96.0
	Frontal	78	3	11	133	63.0	97.8

3.2 *Optimization of Parameters in Consistency Analysis*

We determine the parameters of the proposed algorithm through the manually labeled crosswalk dataset. In this paper, we establish a training and testing dataset to develop the algorithm referring to Ref. 23. The training set includes up to 381 color images, where the quadrilateral boundaries of crosswalks are labeled as ground truth. The crosswalks in the training set have various orientations, sizes, and brightnesses, as shown in Fig. 7.

Precision and recall are usually used as the criteria of detection performance. If all of the detected candidates are inside the ground truth, the detection result is defined as true positive; otherwise, it is defined as false positive. If there is no crosswalk being detected, which is consistent with the ground truth, it is defined as true negative. Comparatively, if the ground truth is positive, it is defined as false negative. For all of the detection results of the dataset, we count the total number of true positive, false positive, and true negative, and define them as TP, FP, and FN, respectively. Herein, the precision and recall of crosswalk detections are defined as

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad \text{and} \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}.$$

Precision reflects the capability to avoid false alarms, and recall denotes the sensitivity of crosswalk detection. To achieve the optimal precision and recall of crosswalk detection, different parameter values are tried on the training dataset. The parameters to be tuned in consistency analysis include the error bounds (E_l, E_a, E_c, E_o, E_{rmin}, and E_{rmax}) and the variance weights (C_1, C_2, C_3, and C_4). An initial parameter combination is assigned to start the tuning procedure, which is shown in Table 3. Different values between the minimum and maximum are tried to find the optimized value of that parameter, where both precision and recall are high.

For the variance weights, a grid search is executed to find four optimized parameters simultaneously. For the error bounds, only one parameter is changing during a tuning course in that the bounds are independent of each other to some degree. After tuning a parameter, the optimized value is updated into the parameter combination, which is prepared for the next tuning. The precision and recall of partial parameter combinations (black points) as well as the fitted receiver operating characteristic (ROC) curve, which is generated by plotting the recall on the vertical axis and 1−precision on the horizontal axis,[26] are presented in Fig. 8.

To achieve optimal performance, the coefficients are determined when crosswalk detection has high precision and recall. There is a trade-off between precision and recall, but in our case, for the crosswalk navigation utility, the precision is of more importance since false alarms are more hazardous than poor sensitivity. Plenty of parameter combinations are eligible for a good performance, and a possible parameter combination is presented in Table 3.

(a) (b)

Fig. 11 (a) Intoer: the wearable navigation system for people with visual impairments. (b) A subject is wearing the navigation system.

Table 5 Voice prompt rules.

Mean orientation range	Bottom of crosswalk	Voice prompt
$o_m < -3$ deg or $o_m > 3$ deg	$b_y < 0.8\ H$	Go ahead
$o_m < -3$ deg	$b_y > 0.8\ H$	Turn left
$o_m > 3$ deg	$b_y > 0.8\ H$	Turn right
-3 deg $< o_m < 3$ deg	$c_x < W/3$	Move to left
-3 deg $< o_m < 3$ deg	$c_x > 2\ W/3$	Move to right
-3 deg $< o_m < 3$ deg	$W/3 < c_x < 2\ W/3$	Cross the road

Table 6 The statistics of field tests.

Subject	Trial times	Detected trials	Aligned trials	Mean alignment time (s)
I	8	6	6	38.3
II	8	5	5	29.5
III	6	4	4	33.8
IV	7	6	5	29.4
V	6	3	3	30.0

3.3 *Performance on Testing Dataset*

We validate the efficiency of the algorithm on the testing set, which is constituted of up to 452 color images derived from nine segments of video. We compare the AECA algorithm with the bipolarity-based algorithm proposed in Ref. 4. The detection results of some typical scenarios using AECA and the bipolarity-based algorithm are shown in Figs. 9 and 10, respectively. Figs. 9(a)–9(c) and 10(a)–10(c) show crosswalk detection results at far distances, at close distances and in front of users, respectively. The results in Fig. 9 illustrate that the crosswalks at different distances can be detected effectively by the AECA algorithm. Nevertheless, using the bipolarity-based algorithm, the crosswalks at far distances fail to be detected and the crosswalks at close distances are detected incompletely, as shown in Fig. 10. Figures 9(d) and 10(d) present the crosswalk detection results in the scenarios of low contrast, which means the gray value of crosswalk stripes is close to that of the background. It can be seen that the AECA algorithm performs well, while it is difficult for the bipolarity-based algorithm to detect crosswalks in such scenarios. In Figs. 9(e) and 10(e), the detection results, where pedestrians or obstacles occlude the crosswalk, are presented. Compared with the ineffective performance of the bipolar algorithm, the crosswalks are detected by the AECA algorithm.

Table 7 The field tests under different testing scenarios.

Scenarios	Trial times	Detected trials	Aligned trials	Mean alignment time (s)
Daytime	31	22	21	32.4
Night	4	2	2	32.5
Sunny day	21	15	15	33.3
Rainy day	10	7	7	30.4
Polyline crosswalks	15	15	14	28.6

Fig. 12 Aligning users with crosswalks from the far to near.

Fig. 13 Crosswalk detection results of field tests.

As shown in Table 4, our algorithm achieves higher precision and recall, especially under the scenarios of occluded crosswalks and far crosswalks. The bipolarity-based algorithm has relatively low precision and recall, because it is sensitive to the predetermined segmenting size of patches. Therefore, the crosswalk, whose size is not matched with predetermined patch size, cannot correctly be detected.

4 Interactive Approach and Field Tests

4.1 *Interactive Approach*

For people with visual impairments, the desired aid system should not only detect crosswalks from images, but also convey the orientation and position of crosswalks to users, so as to instruct them to align themselves with crosswalks. Different auditory interfaces for guiding people with visual impairments during crossing roads are discussed in Ref. 27, and a speech-based interface is one of the best guiding modes. As shown in Fig. 11, as the wearable navigation system used in this paper, Intoer (commercially available at Ref. 28) is composed of a camera, a pair of bone-conducting earphones, and a portable computer.[3] Voice prompts from the bone-conducting earphone, which does not block the hearing completely such as an ordinary earphone,[27] are utilized as the interactive approach in the wearable system.

Similar to the interaction paradigm proposed in Ref. 29, we define three typical crosswalk-interactive paradigms: crosswalks at far distances from the user, crosswalks at close distances from the user, and crosswalks in front of the user, which correspond with three scenarios in Sec. 1. In Fig. 2(a), the crosswalk is at far distances from the user, and the system informs the user to keep walking forward. In Fig. 2(b), the crosswalk is close to the user, and the system informs the user to turn right or left, so as to face to the crosswalk. In Fig. 2(c), the crosswalk is in front of the user, and the system informs the user to move right or left, so as to align himself/herself with the crosswalk. According to different crosswalk detection results, we set corresponding voice prompts. The mean orientation of crosswalk stripes (o_m), the bottom of the crosswalk in image (b), and the center of crosswalk (c) are utilized to determine the suitable prompt. The detailed rules are listed in Table 5.

4.2 *Field Tests*

The portable PC with an Intel Atom x5-Z8500 processor and 2 GB memory is chosen as the computing platform.[30] The resolution of acquired color images needs to be moderate, since sufficient processing speed should be guaranteed on the portable PC. However, images with limited resolution may make crosswalks at far distances unnoticeable. Therefore, in our case, the resolution is set to 1280×720.

To evaluate the performance of the detection algorithm and interactive approach comprehensively, we carried out field tests in another city, which is different from the city of the dataset. Up to 35 trials were completed by five subjects, whose eyes were covered during the test. Starting at 10 m away, they followed the voice prompts to

Table 8 The incomplete detection results and alignment time.

Detection result	Trial times	Mean alignment time (s)
Complete	15	28.2
Incomplete	9	32.4

align themselves with crosswalks. When the subject thinks he or she has aligned with crosswalks, the test is ended. If the subject is facing the walking orientation of crosswalks and within the region of crosswalks, the alignment is successful and the alignment time is recorded. Otherwise, the alignment fails. During tests, crosswalk detection results are observed. If the crosswalks are detected steadily, the test is seen as a detected trial. The statistics of field tests are shown in Table 6.

Out of 35 tests, crosswalks in 24 trials are correctly detected, and nearly all subjects align themselves with crosswalks correctly if crosswalks are detected. The mean alignment time for all of the subjects is 32 s. As shown in Fig. 12, each column presents a field test at an intersection. The three rows of images from top to bottom present the crosswalks at far distances, at close distances and in front of users, respectively.

We carried out the field test in different environments, including daytime, night, a sunny day and rainy day, and the results per testing condition are shown in Table 7. The tests confirm the validation of the detection algorithm and interactive approach. Polyline crosswalks could be detected by our algorithm, as shown in Table 7 and Figs. 12 and 13.

On the experimental platform, the frame rate of crosswalk detection is 15 to 30 fps, which is sufficient for blind assistance usage. More crosswalk detection results of field tests are presented in Fig. 13.

The main problem for the algorithm is that not all of the crosswalk stripes are included in the detection results. Although it does not affect the precision and recall of the algorithm, it affects the performance of the interactive approach. As shown in Table 5, when the crosswalk detection result is close to the bottom of the image, the voice prompt instructs the user to turn right or left, so as to face the crosswalk. Due to the incomplete detection results, the close crosswalk stripes may miss, which results in misleading voice prompts. In the field test, the users spent more time to align themselves with crosswalks, as shown in Table 8.

5 Conclusion

In this paper, we implement the AECA algorithm and the interactive approach on a wearable device to help people with visual impairments cross roads. The experiment proves that the proposed detection algorithm achieves better precision than the conventional algorithm in different scenarios. On the testing dataset, the precision is higher than 80% along with the recall higher than 60%. We have accomplished several challenges: crosswalk detection at far distances, pedestrian occlusion, low-contrast crosswalks, various illuminances, and the limited resources of the portable PC. The field tests illustrate the valid utility of the interactive approach as well as a good frame rate of 15 fps.

In future work, the algorithm will be improved to achieve complete crosswalk detection. Furthermore, intersection recognition could be developed for comprehensive assistance on navigation.

References

1. R. Cheng et al., "A ground and obstacle detection algorithm for the visually impaired," in *IET Int. Conf. on Biomedical Image and Signal Processing*, pp. 1–6 (2015).
2. K. Yang et al., "A new approach of point cloud processing and scene segmentation for guiding the visually impaired," in *IET Int. Conf. on Biomedical Image and Signal Processing*, pp. 1–6 (2015).
3. K. Yang et al., "Expanding the detection of traversable area with real sense for the visually impaired," *Sensors* **16**(11), 1954 (2016).
4. M. S. Uddin and T. Shioyama, "Bipolarity and projective invariant-based zebra-crossing detection for the visually impaired," in *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR 2005)*, p. 22 (2005).
5. J. Coughlan and H. Shen, "A fast algorithm for finding crosswalks using figure-ground segmentation," in *2nd Workshop on Applications of Computer Vision in Conjunction with ECCV* (2006).
6. V. Ivanchenko, J. Coughlan, and S. Huiying, "Detecting and locating crosswalks using a camera phone," in *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition Workshops*, pp. 1–8 (2008).
7. K. Karacs et al., "A mobile visual navigation device: new algorithms for crosswalk and pictogram recognition," in *2nd Int. Symp. on Applied Sciences in Biomedical and Communication Technologies*, pp. 1–2 (2009).
8. M. Hödlmoser, B. Micusik, and M. Kampel, "Camera auto-calibration using pedestrians and zebra-crossings," in *IEEE Int. Conf. on Computer Vision Workshops (ICCV Workshops)*, pp. 1697–1704 (2011).
9. D. Ahmetovic et al., "ZebraRecognizer: efficient and precise localization of pedestrian crossings," in *22nd Int. Conf. on Pattern Recognition*, pp. 2566–2571 (2014).
10. T. Asami and K. Ohnishi, "Crosswalk location, direction and pedestrian signal state extraction system for assisting the expedition of person with impaired vision," in *10th France-Japan/8th Europe-Asia Congress on Mecatronics (MECATRONICS)*, pp. 285–290 (2014).
11. S. Wang et al., "RGB-D image-based detection of stairs, pedestrian crosswalks and traffic signs," *J. Visual Commun. Image Represent.* **25**(2), 263–272 (2014).
12. Y. Wei, X. Kou, and M. C. Lee, "A new vision and navigation research for a guide-dog robot system in urban system," in *IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, pp. 1290–1295 (2014).
13. S. Fontanesi et al., "Real-time pedestrian crossing recognition for assistive outdoor navigation," *Stud. Health Technol. Inf.* **217**, 963–968 (2015).
14. 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 *Int. Conf. on Image Analysis and Processing*, pp. 282–289 (2015).
15. Y. Zhai et al., "Crosswalk detection based on MSER and ERANSAC," in *IEEE 18th Int. Conf. on Intelligent Transportation Systems*, pp. 2770–2775 (2015).
16. M. Khaliluzzaman and K. Deb, "Zebra-crossing detection based on geometric feature and vertical vanishing point," in *3rd Int. Conf. on Electrical Engineering and Information Communication Technology (ICEEICT)*, pp. 1–6 (2016).
17. S. Mascetti et al., "ZebraRecognizer: pedestrian crossing recognition for people with visual impairment or blindness," *Pattern Recognit.* **60**, 405–419 (2016).
18. M. Radványi and K. Karacs, "Navigation through crosswalks with the bionic eyeglass," in *3rd Int. Symp. on Applied Sciences in Biomedical and Communication Technologies (ISABEL 2010)*, pp. 1–2 (2010).
19. S. Wang and Y. Tian, "Detecting stairs and pedestrian crosswalks for the blind by RGBD camera," in *IEEE Int. Conf. on Bioinformatics and Biomedicine Workshops (BIBMW 2012)*, pp. 732–739 (2012).
20. C. Grana, D. Borghesani, and R. Cucchiara, "Optimized block-based connected components labeling with decision trees," *IEEE Trans. Image Process.* **19**(6), 1596–1609 (2010).
21. A. W. Fitzgibbon and R. B. Fisher, "A buyer's guide to conic fitting," in *Proc. of the 6th British Conf. on Machine Vision*, Vol. 2, pp. 513–522, BMVA Press, Birmingham, United Kingdom (1995).
22. M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM* **24**(6), 381–395 (1981).
23. R. Cheng, "Crosswalk data set," http://www.wangkaiwei.org/file/crosswalk%20DATASET.rar (27 June 2017).
24. M. Riedmiller and H. Braun, "A direct adaptive method for faster back propagation learning: the RPROP algorithm," in *IEEE Int. Conf. on Neural Networks*, Vol. 581, pp. 586–591 (1993).
25. N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst. Man Cybern.* **9**(1), 62–66 (1979).
26. C. D. Brown and H. T. Davis, "Receiver operating characteristics curves and related decision measures: a tutorial," *Chemom. Intell. Lab. Syst.* **80**(1), 24–38 (2006).
27. S. Mascetti et al., "Sonification of guidance data during road crossing for people with visual impairments or blindness," *Int. J. Hum. Comput. Stud.* **85**, 16–26 (2016).
28. K. Vision, "Intoer: auxiliary glasses for people with visual impairments," http://www.krvision.cn/cpjs/ (13 September 2017).
29. D. Ahmetovic, C. Bernareggi, and S. Mascetti, "Zebralocalizer: identification and localization of pedestrian crossings," in *Proc. of the 13th Int. Conf. on Human Computer Interaction with Mobile Devices and Services*, pp. 275–284 (2011).
30. Kangaroo, "Kangaroo mobile desktop pro," http://www.kangaroo.cc/kangaroo-mobile-desktop-pro/ (18 December 2016).

Ruiqi Cheng received his BSc degree from Zhejiang University in 2015, and currently he is a master's student at Zhejiang University, China. His current research interests are image processing and machine learning on blind-assisting technology.

Kaiwei Wang received his BSc and PhD degrees from Tsinghua University, Beijing, China, in 2001 and 2005, respectively. He joined the Centre for Precision Technologies, University of Huddersfield, in October 2005 as a postdoctoral research fellow. Since 2009, he has been working as an associate professor at Zhejiang University. To date, his research has been primarily concerned on intelligent guide for people with visual impairments.

Kailun Yang received his BSc degree from the School of Optoelectronics, Beijing Institute of Technology, in 2014, and is currently a PhD candidate at the College of Optical Science and Engineering, Zhejiang University. His current research interest includes stereo vision.

Ningbo Long received his master's degree from Tianjin University in 2015, and is currently a PhD candidate at the College of Optical Science and Engineering, Zhejiang University, China. His current research interests are the small and short range radar systems.

Weijian Hu received his BSc degree in optical science and engineering from Zhejiang University, Hangzhou, China, in 2016. He is currently pursuing the PhD in measuring and testing technologies at Zhejiang University. His current research interest is human computer interaction based on sonification for people with visual impairments.

Hao Chen received his BSc degree in optical science and engineering from Zhejiang University, Hangzhou, China, in 2016. He is currently pursuing the master's degree in optical engineering at Zhejiang University. Since 2016, he has been interested in stereo vision, RGB-D sensor, simultaneous localization and mapping, and assisting technologies for people with visual impairments.

Jian Bai received his master's and PhD degrees in 1992 and 1995, respectively, from Zhejiang University, China. Since 1995, he has been working at Zhejiang University. His current research focuses on optical systems and optical measurement.

Dong Liu received his BSc and PhD degrees in 2005 and 2010, respectively, from Zhejiang University, China. He joined the National Aeronautics and Space Administration in 2010 as a postdoctoral research fellow. Since September 2012, he has been working at Zhejiang University. His current research focuses on optical measurement and remote sensing.