Stairways detection based on approach evaluation and vertical vanishing point

Md. Khaliluzzaman and Kaushik Deb*

Department of Computer Science and Engineering (CSE),
Chittagong University of Engineering and Technology (CUET),
Chittagong-4349, Bangladesh
Email: khalil_021@yahoo.co.in
Email: debkaushik99@cuet.ac.bd
*Corresponding author

Abstract: Stair region detection and distance estimation from a stair image are challenging activities to support visually impaired navigation safely in unknown environments. In this paper, a framework is proposed for detecting stair region from a stair image utilising some natural and unique property of a stair. One unique property of them is, every stair step's beginning and ending horizontal edge point intersects with two vertical edge points creating three connected point (TCP). The TCPs are used to validate the stair edge segments and calculate the vertical vanishing point to justify the stair edges. This justification ensures that validated edge segments are arranged in an increasing parallel order which is the other unique property of a stair. These increasing edge segments are verified by utilising the y coordinate value of the vanishing point and the detection of stair candidate region is confirmed by these properties. In addition, the triangular similarity is used for distance estimation from camera to stair. The proposed framework is tested using various stair images under a variety of conditions and results are presented to demonstrate the efficiency and effectiveness.

Keywords: autonomous stair climbing navigation; alarm system; Gabor filter; illumination; vanishing point; three connected point; TCP; triangular similarity.

Reference to this paper should be made as follows: Khaliluzzaman, M. and Deb, K. (2018) 'Stairways detection based on approach evaluation and vertical vanishing point', *Int. J. Computational Vision and Robotics*, Vol. 8, No. 2, pp.168–189.

Biographical notes: Md. Khaliluzzaman received his BSc in CSE in 2007 from Khulna University of Engineering and Technology (KUET), Bangladesh. He received his MSc in CSE in 2017 from Chittagong University of Engineering and Technology (CUET), Bangladesh. Currently, he is working as an Assistant Professor in the Department of Computer Science and Engineering, International Islamic University Chittagong, Chittagong, Bangladesh. His research interests include image processing, computer vision, human computer interaction, and pattern recognition.

Kaushik Deb received his BTech and MTech degrees from Department of Computer Science and Engineering of Tula State University, Tula, Russia, in 1999 and 2000, respectively. He received his PhD degree in Electrical Engineering and Information System from University of Ulsan, Ulsan, South Korea, in 2011. Since 2001, he has been serving as a faculty member in the Department of Computer Science and Engineering (CSE), Chittagong University of Engineering and Technology (CUET), Chittagong, Bangladesh.

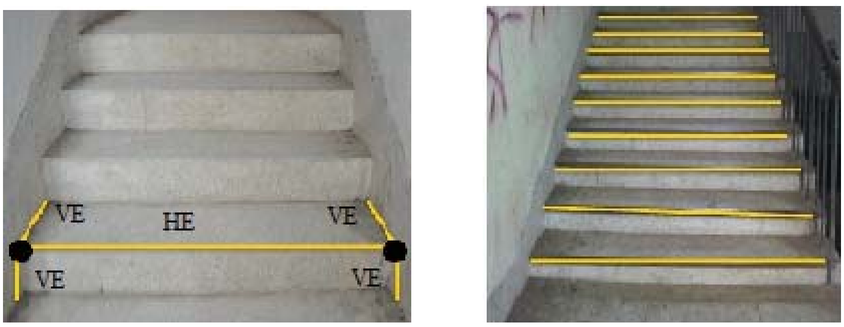
Currently he is a Professor in Department of CSE and Director of the Institute of Information and Communication Technology (IICT), Chittagong University of Engineering and Technology (CUET), Bangladesh. He served as the Head of the Department of Computer Science and Engineering, Chittagong University of Engineering and Technology (CUET), Bangladesh during 2012–2015. His research interests include computer vision, pattern recognition, intelligent transportation systems (ITSs), and human-computer interaction.

This paper is a revised and expanded version of a paper entitled 'Stairways detection and distance estimation approach based on three connected point and triangular similarity' presented at 9th International Conference on Human System Interaction (HSI2016), University of Portsmouth, Portsmouth, UK, 6–8 July 2016.

1 Introduction

The trend of efficient, appropriate, and advanced intelligent autonomous systems (IASs) become popular in recent years. Accurate detection and recognition of a particular object in an image is one of the most challenging tasks in the field of computer vision. Detection and distance estimation of a stair region in a stair image is such a task. The stair region detection and estimation of distance from a camera to stair is widely used for autonomous system and vision impaired people. Detection of stair from stair images is difficult because of unusual shape, change of viewpoints, and non-uniform indoor and outdoor illumination conditions while capturing images. Furthermore, to satisfy the criteria of real-time system and IASs, the stair detection method should operate as fast as possible and detect each interest object from the stair image.

Figure 1 Some natural property of stair, (a) TCP (b) horizontal edges in parallel increasing order (see online version for colours)



(a) (b)

This paper proposes a framework based on natural and unique properties of a stair. One natural property of stair is, every stair step's beginning and ending horizontal edge (HE) point intersects with two vertical edge's (VE) points. One vertical edge is stair step height, and another edge is stair step width. This makes up three connected point (TCP) as shown in Figure 1(a). This is the key part of the proposed framework. Another natural property of a stair is that stair step's horizontal edges appear gradually increasing order from top to bottom of the stair in a parallel arrangement as shown in Figure 1(b). Every stair shows these properties when observed the stair front side from a short distance of the stair similar to the human vision.

Stair image could be in different environments including indoors and outdoors, different illuminations, shadows, etc. In order to deal with these situations, a directional Gabor filter (Daugman et al., 1985; Weldon et al., 1996; Cong et al., 2008; Hernández and Jo, 2010) is applied to eliminate the influence of the illumination. The Canny edge detector (Canny, 1986) method is applied for edge detection. Furthermore, non-candidate stair edges are eliminated by applying a filtering operation. Then the proposed edge linking method is used to extract the longest horizontal parallel edges. Those parallel edges are required to find the TCP at the beginning and ending point of the horizontal step edges of the stair edge image. The TCP's validate the stair step's horizontal parallel edges. This is the key part of the proposed framework. Finally, validated horizontal parallel edges are used to calculate the vertical vanishing point (McLean and Kotturi, 1995; Schaffalitzky and Zisserman, 2000; Hernández and Jo, 2010; Deb et al., 2013) to justify and verify the stair edges and confirm the detection of stair candidate region. After that Pythagoras and triangular similarity are used to estimate the distance from a camera to the stair.

The rest of the paper is organised as follows. In Section 2 presents the related research done by researchers in the past to address this problem. The proposed stair detection framework is described in Section 3. In the next section distance is estimated. The experimental results are explained in Section 5. The paper concludes in Section 6.

2 Related research

This section demonstrates a summary of the various methods that have been tested and implemented for stairway detection and estimation of distance from a camera to the stair in the stair image. This topic has gained importance for the purpose of automatic navigation of autonomous systems. All the available research works focused on the property that an edge image of a stair contains concurrent long horizontal edges. As the detection and distance estimation of stair region is concerned, scientist and researchers have coined extensive methods to improve the stair detection and distance estimation accuracy. For example, Schwarze and Zhong (2015) presented a method which enables a robotic system to perceive the parameters of a stair necessary to traverse a stairway using stereo vision. This approach introduces a light-weight method to measure the individual step of a stair. This information is used to update a minimal stair model while approaching and traversing the stair. Before that, Delmerico et al. (2013) proposed a path planning system for detecting and modelling ascending stairways by a mobile robot. The objective of this system is to map and localise building's multiple floors simultaneously. Therefore, each stairway can be estimated by the physical properties of the stair. Wang et al. (2014) developed another framework based on RGB-D images to detect and recognise stairs, pedestrian crosswalks, and traffic signals. To extract the concurrent parallel lines from an image, the Hough transforms is applied (Duda and Hart, 1972; José et al., 2012). Then the Depth channel is used to recognise pedestrian crosswalks and stairs. Finally, the distance between the camera and stairs is estimated for blind users.

Hernández and Jo (2010) utilised some constraint such as the vanishing point (Barnard, 1983; McLean and Kotturi, 1995; Schaffalitzky and Zisserman, 2000) and the directional Gabor filters (Lee, 1996; Basca et al., 2007; Cong et al., 2008) to localise and recognise outdoor stairways. By utilising these constraints, longest horizontal edge segments are extracted, and finally, detect the stair candidate region. Later, Hernández et al. (2011) presented a localisation method to detect and localise the indoor stairways. This method analyses the stair edges by employing the directional filter and planar motion tracking. Then the horizontal edges are extracted by applying the Gabor filter. From this set of horizontal edge segments, a hypothetical set of edges is extracted by using a correlation method. Finally, a discrimination method is used to detect the ground plane on the basis of behavioural distance measurement. Deb et al. (2013) exploited a natural property of stair to detect the stair region from a stair image. The idea of this framework is that stair steps appear in sorted order from top to bottom of a stair by their length. For that, Sobel's edge detector was applied for extracting horizontal edges, and then longest increasing subsequence algorithm was employed for extracting longest horizontal edges in a parallel arrangement. Finally, detection of stair region is confirmed by calculating the vanishing point (Hernández and Jo, 2010). Zhong et al. (2011) proposed an autonomous navigation system based on Gabor filters and fuzzy fusion phase grouping (FFPG) for stairway climbing. Se and Brady (2000) used a texture detection method based on Gabor filters to detect the distant staircases. When the stairs are close enough, then the staircases are looking like a group of parallel edges according to human vision, where convex and concave edges are portioned using intensity variation information. This method also estimates the poses of stairs by a homograph search model.

Shahrabadi et al. (2013) presented an algorithm for detecting stair region both in indoor and outdoor environments. The algorithm starts with smoothing the image with a Gaussian low pass filter with mask 10×10 and $\sigma = 4$. After that, Canny operator and Hough transform are applied to detect the approximately horizontal edges. Finally, use some criteria such as white canes to detect the stair region. Before that, duBuf et al. (2011) performed a project of Smart Vision for assisting the blind and visually impaired while navigating autonomously. After that, José et al. (2012) presented a real-time vision system for the path and obstacle detection to assist the blind and visually impaired persons. Carbonara and Guaragnella (2014) proposed a stair detection system to help in the development of world awareness for vision impaired people through the Smartphone. Lee et al. (2012) introduced a real-time stair detection framework. This framework uses AdaBoost learning algorithms to detect the stair and estimate the ground plane of a stair.

Cong et al. (2008) proposed a stereo vision-based algorithm for autonomous stair climbing navigation system. This algorithm estimates the parameters using RANSAC algorithm to guide the UGV to climb the stairs automatically under various conditions. Lu and Manduchi (2005) presented another stereo vision system for autonomous navigation to recognise the stair in order to define the actual position of the stairway into a stair image. This system combines the brightness information in the form of edges with 3D data from a commercial stereo system.

This paper proposes a framework based on natural and unique properties of a stair. One of them is, every stair step's beginning and ending horizontal edge point intersects with stair step's height and weight vertical edges point i.e., TCP. Another one is stair step's horizontal edges appears in increasing order from top to bottom of a stair in a parallel arrangement. After detecting the stair candidate region, the triangular similarity is used to estimate the distance from a camera to the stair.

3 Proposed framework

In this section, the proposed stair detection framework has been described in details. The framework is divided into six primary steps, i.e.,

1 applying Gabor filter and detecting stair edges

2 filtering non-candidate edges

3 edge linking and tracking

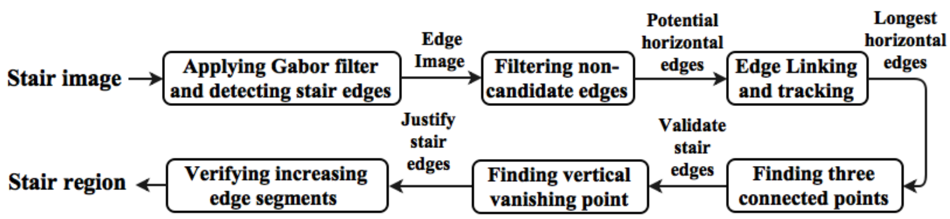
4 finding TCP's

5 calculating vertical vanishing point

6 verifying the increasing horizontal edge segments and detecting stair candidate region.

The proposed framework is as shown in Figure 2.

Figure 2 The proposed stair detection framework



3.1 Applying Gabor filter and detection stair edges

In order to extract the stair edges properly, noises and shadow effect should be removed from the stair image. For that, Gabor filter can be applied using standard convolution method. Initially, the input stair image is converted into a grey scale image for reducing computational cost. A grey scale image has only one channel which contains intensity information for each pixel of the image. Gabor filter is applied to the grey scale image in order to remove the influence of illuminations of the stair image and extract the different orientated stair edges. It is a linear and directional wavelet type filter or mask (Weldon et al., 1996; Basca et al., 2007; Hernández et al., 2011). They consist of a sinusoidal plane wave of a particular frequency and orientation modulated by Gaussian envelope. The Gabor filter is optimum over other filters due to its optimal localisation properties in both spatial and frequency domain.

Usually, stair is located in a place where natural light falls and lighting condition varies at different daytimes. With appropriate selection of Gaussian variance, orientation, and wavelength, the Gabor filter can remove the influence of illumination variety and shadow effects on an image efficiently and reserve the multi-scale and multi-directional information. The appropriate parameter of a Gabor filter is selected to get the best response from the noise and maintain the stair edges of the stair image. In this regard, various sets of values for variance of Gaussian, wavelength, and orientation are chosen to get the most suitable Gabor filter. The general form of a Gabor filter is defined in (1), (2), and (3).

$$G_{\lambda,\theta,\varphi,\sigma,\gamma}(x, y) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right)\cos\left(2\pi\frac{x'}{\lambda} + \varphi\right) \tag{1}$$

$$x' = x\cos\theta + y\sin\theta \tag{2}$$

$$y' = -x\sin\theta + y\cos\theta \tag{3}$$

where, the original coordinate of the pixel is (x, y) and λ represents the wavelength of the cosine factor, θ is the orientation, φ is the phase offset, σ is the standard deviation of the Gaussian envelope, and γ is the spatial aspect ratio. In the (x, y) coordinates, the Gabor filter can only be stretched alone the x-axis or the y-axis. However, it does not stretch diagonally. Equations (2) and (3) allow the Gabor filter to be stretched in any orientation defined by θ in the (x', y') coordinate.

The standard deviation σ is not independent, it depends on bandwidth (b) and wavelength (λ). The relation between b and λ is as shown in (4).

$$\sigma = \frac{1}{\pi}\sqrt{\frac{1n2}{2}\frac{2^b + 1}{2^b - 1}}\lambda \tag{4}$$

If the bandwidth is 1 the relation of σ and λ is $\sigma = 0.56\ \lambda$. In this work, it is assumed that all stair edges are horizontal or nearly horizontal and the bandwidth is equal to 1. Figure 3(b) shows the result after applying Gabor filter on the input stair image.

According to the natural properties of a stair, the stair steps are horizontally parallel to each other. Hence, the horizontal edges convey information of stair. So, all horizontal edges from the stair image need to be extracted and the potential stair edges need to be found out. Another natural property of stair is that every stair step's beginning and ending horizontal edge point intersect with TCP. So, the vertical edges from the stair image also need to be extracted. The Canny (1986) edge detector is applied on the filtered grey scale image to extract the edges.

Figure 3 Processing example of detecting stair edges, (a) stair image (b) result of after applying Gabor filter (c) canny edge image (see online version for colours)

(a) (b) (c)

Canny operator which is the Laplacian operator (Chen, 2012) is chosen in this work because of its better performance over Gradient operator such as Roberts, Prewitt, and Sobel (Maini and Aggarwal, 2009; Shrivakshan and Chandrasekar, 2012). The sensitivity of the Gradient operators over noise is very high. The dimension of a Gradient operator's kernel filter and its coefficients are static. It cannot be adapted with a given image. Whereas, the Laplacian operator such as Canny operator is adaptable with different noise level, and has the concept of smoothing. It improves the signal to noise ratio by non-maximum suppression method. It also detects the edges efficiently under noise with the help of thresholding method. The main advantage of Canny operator is that it has the adjustable parameters. The parameters are standard deviation for the Gaussian filter, and its threshold values (Huertas and Medioni, 1986). The edge detection result is as shown in Figure 3(c).

3.2 Filtering non-candidate edges

In this section, potential edges are extracted from the stair edge image. At this stage, the edge image contains different oriented edges. To extract all horizontal edges from the edge image, at first the vertical edges are eliminated from the edge image. After completing the present stage, the edge image mainly contains the horizontal edges. Figure 4(a) shows the result of eliminating vertical edges from the Canny edge image.

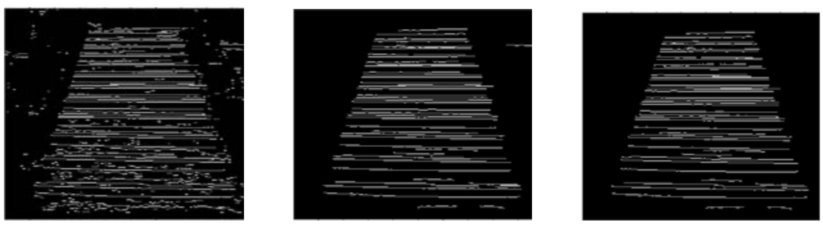
Besides these, a lot of small and noise edges exist in the edge image. In order to filter the small edges that are not needed, in this paper, a method has been proposed. The idea of this method is that if an edge line is bigger than the THRESHOLD_LINE then the bigger line will be preserved. The procedure to compute the THRESHOLD_LINE size is, length (i) = length of edge i and THRESHOLD_LINE = max (length (i)) /6.

Small edge eliminating procedure

```
for i = 1 to No_of_edge
If length (i) >THRESHOLD_LINE then
    Preserve the edge i pixels
else
    Eliminate the edge i pixels
end if
end for
```

This method eliminates all small and discontinuous edges and saves time for edge linking. Figure 4(b) shows the result of eliminating small edges from the edge image. Still in the edge image many edges exist that are not part of the parallel concurrent edge. Those edges are eliminated from the edge image as a non-candidate edge, as shown in Figure 4(c).

Figure 4 Processing example of small edge elimination, (a) horizontal edge image (b) eliminate small edges (c) eliminate non-candidate edges

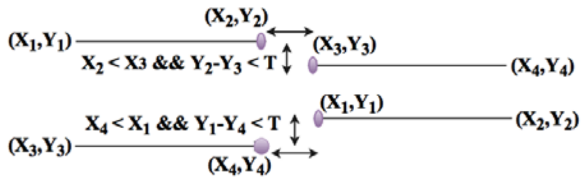


(a) (b) (c)

3.3 Edge linking and tracking

At this stage, the edge image contains candidate stair's long horizontal edges only. The long horizontal edges may have small breaks or gaps at some places for various reasons. Still, these pieces represent one horizontal edge. So, small breaks or gaps are required to be filled up. For that, initially horizontal gap of 3 pixels or less are automatically filled up. If the gap size is more than 3 pixels, then edge linking procedure will be applied. By this procedure, the breaks or gaps in the horizontal edges are efficiently filled in the edge image. Figure 5 shows the process of edge linking and tracking.

Figure 5 Edge linking and tracking (see online version for colours)



The procedure for edge linking and tracking is given below:

Edge linking and tracking procedure

---

x_l^i, x_r^i = x coordinate of the i[th] edge's left and right end point

y_l^i, y_r^i = y coordinate of the i[th] edge's left and right end point

Step 1 Arrange the horizontal edge segments in descending order by the value of y_l^i

Where i = 1,…,N and N is the total number of edge line

Step 2 The merging process

for i = 2 to N

If $y_r^i - y_l^{i-1}$ < threshold and $x_r^i - x_l^{i-1}$ then

merge edge i[th] and (i–1)[th] by tracking edge

point (x_r^i, y_r^i) and (x_l^{i-1}, y_l^{i-1})

else if $y_l^i - y_r^{i-1}$ < threshold and $x_l^i > x_r^{i-1}$ then

merge edge i[th] and (i–1)[th] by tracking edge

point (x_r^{i-1}, y_r^{i-1}) and (x_l^i, y_l^i)

else continue

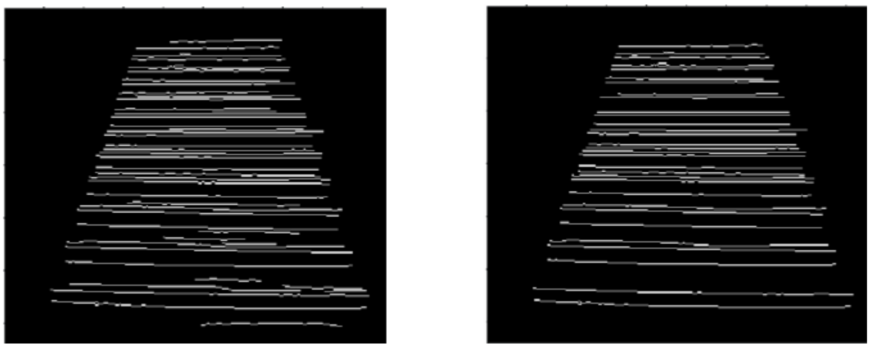
end if

end for

Here, the threshold value is 3.

The result of applying edge linking and tracking procedure in an edge image is shown in Figure 6(a). Finally, those edges are removed from the edge image whose lengths are less than a threshold is shown in Figure 6(b). Here max(length(i))/5 is used as this threshold value. This step ensures that, the edge image contains the potential long horizontal edges only. Let the number of extracted horizontal edge is N.

Figure 6 Processing example of edge linking, (a) edge linking (b) potential longest horizontal edges



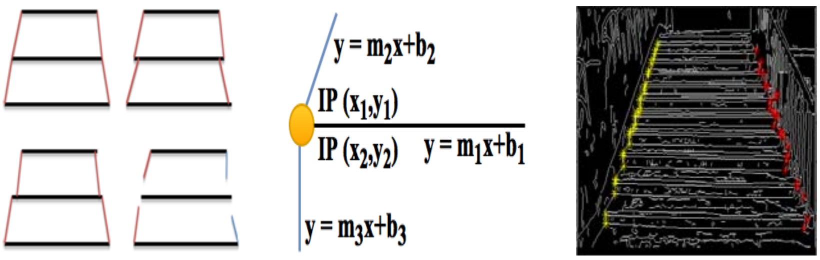
(a) (b)

3.4 Finding TCP's

This section focuses on detecting TCP's from the edge image. This is done by utilising a unique property of stair, that is, the beginning and ending points of stair step's horizontal edges intersect with two vertical edge points. These vertical edges are from stair step's height and its width, as shown in Figure 1(a). These TCP's validate the extracted horizontal edges are from stair step edge. For this task, Canny edge image has been used from where longest N horizontal edges are extracted and it also contains vertical edges. According to this property, vertical edges, i.e., stair step height and its width intersect with horizontal edge's starting and ending points. These intersection points are at the same coordinate of horizontal edge's starting and ending points.

For that, the stair step's height and width vertical edges are searched at the beginning and ending point of the longest horizontal edges for finding the intersection points. If the intersection point is found at the beginning side of the horizontal step edge, then another intersection point is searched with vertical edges at the ending coordinate. In the practical situation, all horizontal and vertical edges in the edge image are not exactly horizontal or vertical. So it is difficult to get the intersection point at the ending point of the horizontal or vertical edges exactly, as shown in the Figure 7(a). For that purpose, the intersection point is detected with a procedure as shown in Figure 7(b).

Figure 7 Processing example of TCP, (a) different orientation of stair step's height and width edges (b) procedure of calculating TCP (c) TCP's in the edge image (see online version for colours)



(a) (b) (c)

According to the procedure shown in Figure 7(b), if $x_1 = x_2$ and $y_1 = y_2$ then the intersection point (IP) will be (x_1, y_1). However, all times the (x_1, y_1) and (x_2, y_2) would not be the same because the horizontal and vertical edges are not exactly horizontal or vertical. For that purpose, we will consider x and y coordinate displacement from the starting or ending point of the horizontal edge with a threshold value. The threshold value of x coordinate is $x\_threshold = 5$ and y coordinate is $y\_threshold = 2$. According to that, if $x_2 - x_1 < x\_threshold$ and $y_2 - y_1 < y\_threshold$ then the average of y_1 and y_2 is taken as y and the intersection point will be (x_1, y). Figure 7(c) shows the TCP's of the vertical and horizontal edges both in the beginning and ending side of the horizontal edges. According to this procedure, checking TCP's at the two ends of the horizontal edge requires constant time, say, c. Hence, finding two TCP at the two end points of a horizontal edge takes 2c time. Therefore, finding TCP's for N horizontal edges need 2Nc time i.e., time complexity arises about $O(N)$. The approximate numbers of stair steps are measured after completing the detection of TCP's.

In the previous stage, the N horizontal edges were extracted as stair steps candidate edge. If the approximate number of stair steps with TCP's in the stair image is 70% with respect to N horizontal edges, then the extracted N edges are partially considered as stair edge. These longest potential N horizontal edge segments will be used in the next section for confirming and detecting stair region shown in Figure 8(a).

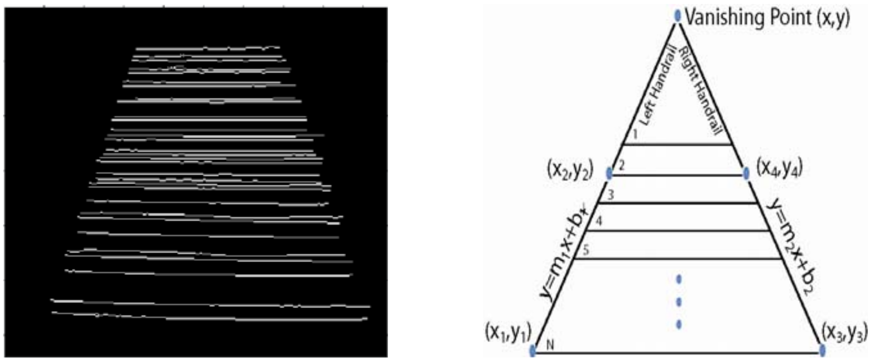
3.5 Calculating vertical vanishing point

This section considers calculating the vertical vanishing point (VP) from the long horizontal N edges that are found in the previous section and validated by the TCP. The vanishing point is used to justify the N horizontal edges, whether these edges are extracted from stair or any other similar visual patterns. The VP of a stair can be stated as an imaginary point in a linear perspective drawing in which a set of parallel lines are converged. In stairways, the VP can be defined as a point where two handrails of the stair intersect. Basically, the vanishing point (x, y) is located outside the stair image. However, every stair does not have the handrails or some stair have one side handrail. Hence, it is better to construct two virtual handrails instead of depending on real handrails. The right and left handrail is the straight line passing through the TCP's on the right and left side of the horizontal edge segments respectively. Hence, the VP is the intersection point of these two left and right straight lines.

In Figure 8(b), the left drawing handrail is the straight line passing through left side's TCP's such as (x_1, y_1) and (x_2, y_2). The equation of the straight line is $y = m_1x + b$ where, $m = (y_2 - y_1) / (x_2 - x_1)$. Similarly, equation $y = m_2x + b_2$ is derived from right handrail passing through right side's TCP's such as (x_3, y_3) and (x_4, y_4), where, $m_2 = (y_4 - y_3) / (x_4 - x_3)$. Solving these two equations, the intersection point (x, y) is determined which is the vanishing point of the stair.

If the vanishing point is determined from these horizontal parallel edges that are validated by the TCP's, then these horizontal edges become smaller from bottom to top until these edges are converged. That means, these horizontal edges gradually appear in an increasing order from top to bottom of the stair in the parallel arrangement. This is another unique property of the stair. If any object edges are in an increasing sequential order, then these edges may be considered as a staircase. Any object containing some long horizontal texture may produce long horizontal edges in its edge image. However, these edges will not appear in an increasingly sorted order.

Figure 8 Processing example of extracting longest increasing horizontal edges, (a) longest horizontal edge segments (b) calculating vertical vanishing point (see online version for colours)



(a) (b)

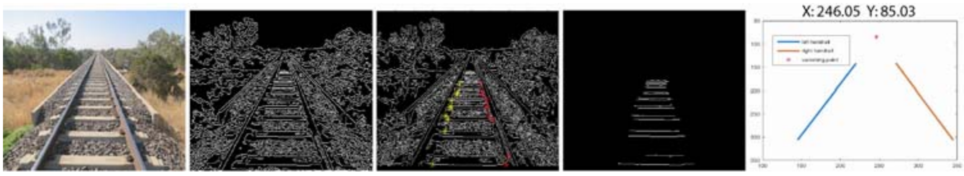
3.6 *Verifying the increasing horizontal edge segments*

In this section, the increasing horizontal edge segments are used to verify the stair's horizontal edge segments that are justified in the previous section. For that, the *y* coordinate of the vanishing point is used and checked to confirm whether these edges are from stair or other similar visual pattern such as train line or zebra-crossing. Basically, the vanishing point of a stair will be imaginary ($y < 0$) and will reside inside the range of $-2IH \leq YVP < 0$ where, *IH* is the height of the image and *YVP* is the *y* coordinate value of the vanishing point. If the calculated VP resides inside the range, the extracted parallel edge segments certainly indicate a stair. On the other hand, those edge segments may be part of some other stair looking object, such as zebra-crossing and rail line. They have the similar property as like stairs. The zebra-crossing and rail line edges are fully horizontal as like stair steps horizontal edges. However, the *y* coordinate value of the vanishing point both of the zebra-crossing and rail line will be real.

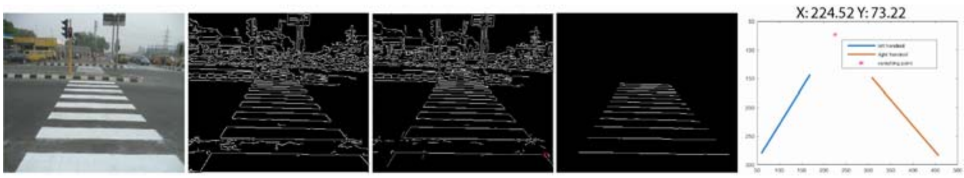
Figure 9 shows the experimental result of a rail line, i.e., sample 1 and zebra-crossing i.e., sample 2 images. In sample 1, an extracted horizontal edge segment satisfies the TCP criterion for rail line image which is the property of stair edges. However, the calculated vanishing point is (246.05, 85.03) as shown in Figure 9(e). The *y* coordinate of the VP is 85.03. The positive *y* coordinate value distinguishes this image from a stair image. Sample 2 shows the zebra-crossing image, that does not satisfy the TCP criterion. Rail line and zebra-crossing VP reside inside of the image as they do not have any slope with respect to the ground plane. So, the *y* coordinate of the VP of the rail line and the zebra-crossing is positive.

However, stair has the slope with respect to the ground plane, for that reason stair's VP resides outside of the stair image. Figure 10(a) shows the example of calculating VP of a stair where the *y* coordinate of the VP is –163.30. Figure 10(b) shows the stair candidate region in the original stair image. The detected stair candidate region is as shown in Figure 10(c).

Figure 9 Processing example of stair like object rail line and zebra-crossing, (a) input image (b) extracted edges (c) TCP's (d) longest increasing horizontal edges (e) vertical vanishing point (see online version for colours)



Sample 1: rail line



Sample 2: zebra-crossing

(a) (b) (c) (d) (e)

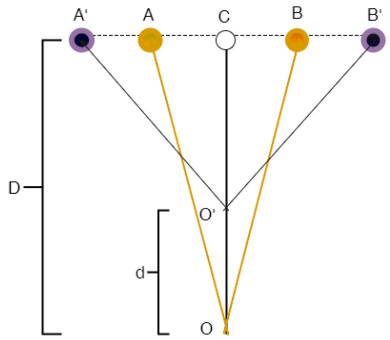
Figure 10 Processing example of detecting stair region, (a) vertical vanishing point (b) detected ROI (c) stair candidate region (see online version for colours)

(a) (b) (c)

4 Estimating distance from camera to stair

Estimating distance from a camera to stair's first step is essential for blind and visually impaired people. By estimating distance they should maintain their walking speed and foot height. It also improves the climbing process of autonomous system in an unknown environment. To estimate the distance from a camera to stair first step, two cameras have been considered in this work. The cameras are placed at the fixed distance with the same height from the ground. The procedure is as shown in Figure 11. The original distance is estimated from the camera placed at O position. It acts as the first camera. The camera placed at O' is used for calibration. It acts as the second camera in this framework. At first, the beginning and ending point of the stair's first step edge is estimated from the camera position at O'. The estimated points are $A'(x_1, y_1)$ and $B'(x_2, y_2)$. The estimated distance between the two points is $A'B' = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$. If the centre point of the two estimated points is C then the distance from the left point to the center point of the edge is $A'C = A'B'/2$. The first camera is placed at O from where the distance of the stair first step will be estimated. From camera at O the estimated position of the beginning and the ending point of the stair's first step edge are A and B. The distance between A and B is AB and the center point of AB is also at C. Let us assume that, the distance between the two cameras is $OO' = d$, the distance from the center point C to the camera at O and O' is $OC = D$ and $O'C = D - d$ respectively.

Figure 11 The proposed framework for estimating distance from the camera (see online version for colours)



From Figure 11, it is seen that $\Delta A'CO'$ and ΔACO are similar triangles. And the relation between the distance from the centre point of stair's first step edge to camera i.e., CO' or CO and the centre point to the left end point of stair's first step edge i.e., CA' or CA is inversely proportional. That is,

$$CO' \infty \frac{1}{CA'} \text{ or } CO \infty \frac{1}{CA} \quad (5)$$

By using $\Delta A'CO'$ and ΔACO similarity, and inverse relation in (5) we can find,

$$\frac{OC}{A'C} = \frac{O'C}{AC} \quad (6)$$

Here, $OC = D$, $O'C = D - d$

$$\frac{D}{A'C} = \frac{D-d}{AC}$$

$$D = \frac{d}{1 - \frac{AC}{A'C}} \quad (7)$$

$$D = \frac{d}{1-\alpha} \quad (8)$$

where α is the ratio of AC and $A'C$, d is the distance between the two cameras. $A'C$ and AC is the half of the distance of $A'B'$ and AB. The estimated distance between the two end points of the first stair step edge is $A'B'$ and AB that can be found through the image processing. By using similarity property the distance from the camera to the first stair step can be estimated, as described in (8).

4.1 Implementation

According to (7), during implementation, the cameras are placed at the fixed distanced. The distance d should not be too long or too short. If the d is too long or too short, the linearity relation of AC and $A'C$ could not be maintained. In this work, the distance between the two cameras is considered as $d = 0.05$ m. The cameras should be placed in front of the stair and the height of the cameras from the ground should be less than 2 metres. The angle orientation of the camera should be as closed as the horizontal axis. The framework for distance estimation is as shown in Figure 12. And the camera setup is as shown in Figure 13. Due to the shortage of the equipment the images are captured using only one camera moving on a track.

Figure 12 The proposed framework for distance estimation

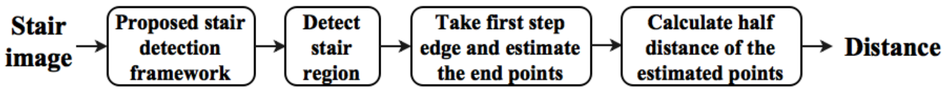


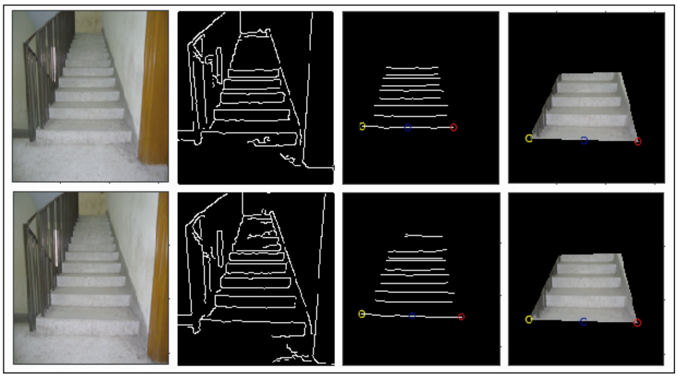
Figure 13 The camera setup, (a) camera with experimental prototype (b) camera with an experimental prototype at the time of capturing stair image (see online version for colours)

(a) (b)

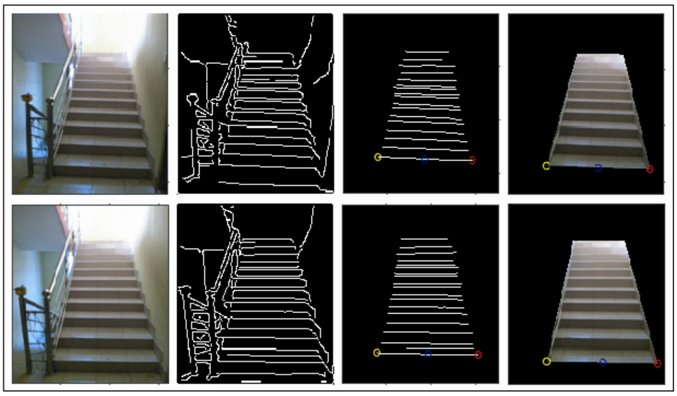
After detecting the stair candidate region, the distance between the camera and the stair's first step is estimated. The processing example is as shown in Figure 14. The top and bottom stair image of each sample in Figure 14(a) is captured from the camera placed at O and O' respectively. The beginning, ending, and the centre point of the stair's first step edge are estimated from the first and the second camera. From these estimated points AC and $A'C$ is calculated. Finally, the distance is estimated by (8).

In Figure 14, stair sample image 3 and image 4 are captured from the indoor environment and sample image 5 is captured from the outdoor environment in a different illumination condition for distance estimation. These stair samples are captured from different distances. The distance estimation accuracy from different stair sample images is as shown in Table 4.

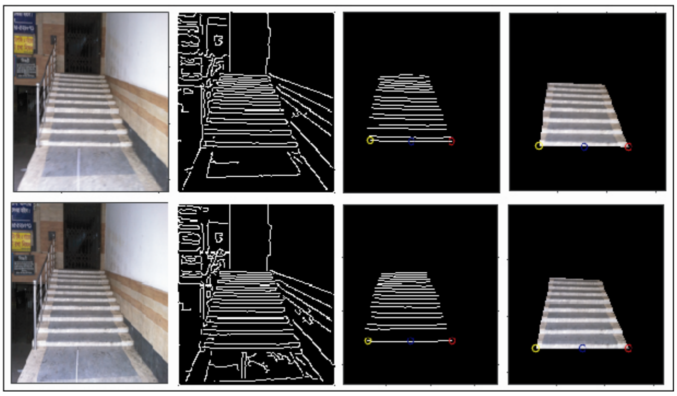
Figure 14 Processing example of distance estimation, (a) stair image (b) stair edge image [(c) and (d)] the beginning, ending and the centre estimating point of first stair step edge in the horizontal edge image and the detect stair region (see online version for colours)



Sample 3



Sample 4



Sample 5

(a) (b) (c) (d)

5 Experimental results

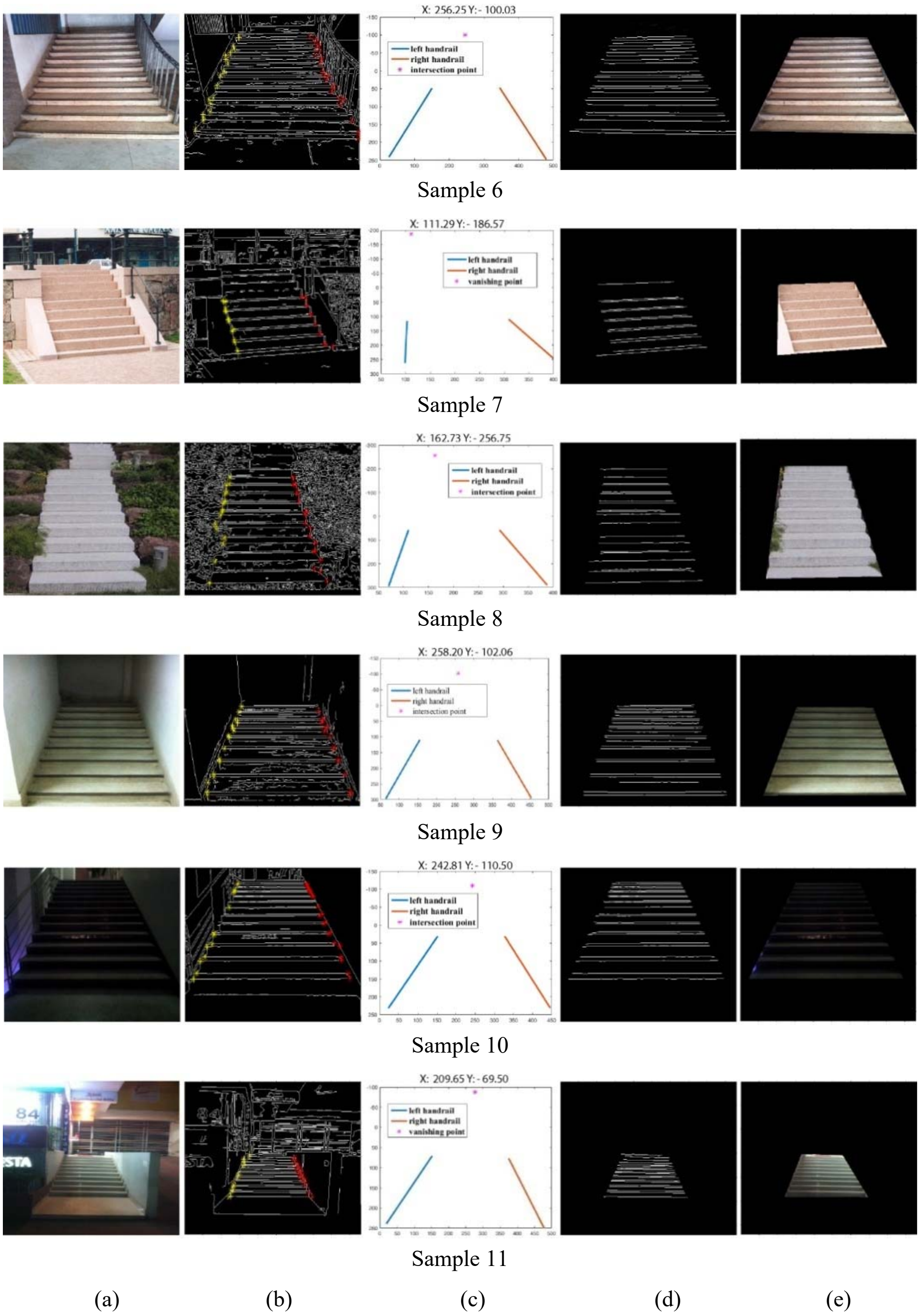
This section shows the experimental results of various real stair images. All experiments were done on an Intel(R) Core(TM) i3-3120M at 2.5 GHZ processor with 4 GB RAM. In this paper, all the processing was done on 480 × 320 stair images. The processing was implemented using MATLAB environment. In the experiments, 105 stair images were used to consider various types of outdoor and indoor stair images. These images are captured from the different environment at various illumination conditions such as normal, uneven, and noisy background as shown in Figure 15. However, some criteria should be maintained during capturing of the stair image. The camera should be placed on the front side of the stair. The image should be captured at a height that should be less than 2 metres from the ground, and not so far away from the stair. The angular orientation of the camera should be as closed as the horizontal axis.

Figure 15 Various types of stair image, (a) indoor stair with uneven illumination (b) outdoor stair with uneven illumination (c) outdoor stair image with noisy background (see online version for colours)

(a) (b) (c)

Figure 16 shows the processing example of stair sample images at different environment and illumination conditions. Input stair image, TCP's, vertical vanishing point, increasing horizontal edges, and the corresponding output of the processing sample images are shown in Figure 16(a), Figure 16(b), Figure 16(c), Figure 16(d), and Figure 16(e), respectively. In Figure 16, stair sample 6 and sample 7 are straight forward indoor and outdoor stair images captured at day time with the normal illumination condition. Stair sample 8 shows the result on an outdoor stair with noisy background at day time where some part of stair region is cropped. Stair sample 9 shows the experiment on the indoor stair at day time with a noisy background. Stair sample 10 shows the result on the indoor stair at lower illumination condition, and stair sample 11 shows the experiment on the outdoor stair at uneven illumination condition.

Figure 16 Processing example for extracting stair region, (a) input image (b) TCP's in edge image (c) vertical vanishing point (d) increasing horizontal edges (e) stair candidate region (see online version for colours)



In Figure 16, stair sample 8 is longest stair compared to other stair samples. In this sample, the beginner stair edges are detected efficiently. After a certain distance, the stair edges are not extracted properly because of a further portion of the sample 8 is narrow, depth, and edges are not resolved with respect to the front edges. Each of these stair samples has satisfied the condition of TCP's, increasing horizontal edges (HE), and having a VP with a negative y coordinate value. These conditions and run time of stair samples are shown in Table 1. The proposed framework detects the stair successfully from all these varieties of stairs.

Table 1 TCP's, vertical vanishing point and run time of stair sample images

Stair sample	*Total HE*	*Total TCP*	*TCP (%)*	*Vertical vanishing point*	*Run time (s)*
Sample 6	24	20	83.33	(256.25, –100.03)	0.061
Sample 7	15	11	73.33	(111.29, –186.57)	0.068
Sample 8	17	14	82.35	(162.73, –256.75)	0.071
Sample 9	22	19	86.36	(258.20, –102.06)	0.065
Sample 10	23	18	78.26	(242.81, –110.50)	0.069
Sample 11	25	23	92.00	(209.65, –69.50)	0.070

Figure 17 Processing example for extracting stair region in different orientations, (a) input image (b) edge image (c) TCP's in edge image (d) longest increasing horizontal edge segment (e) stair candidate region (see online version for colours)

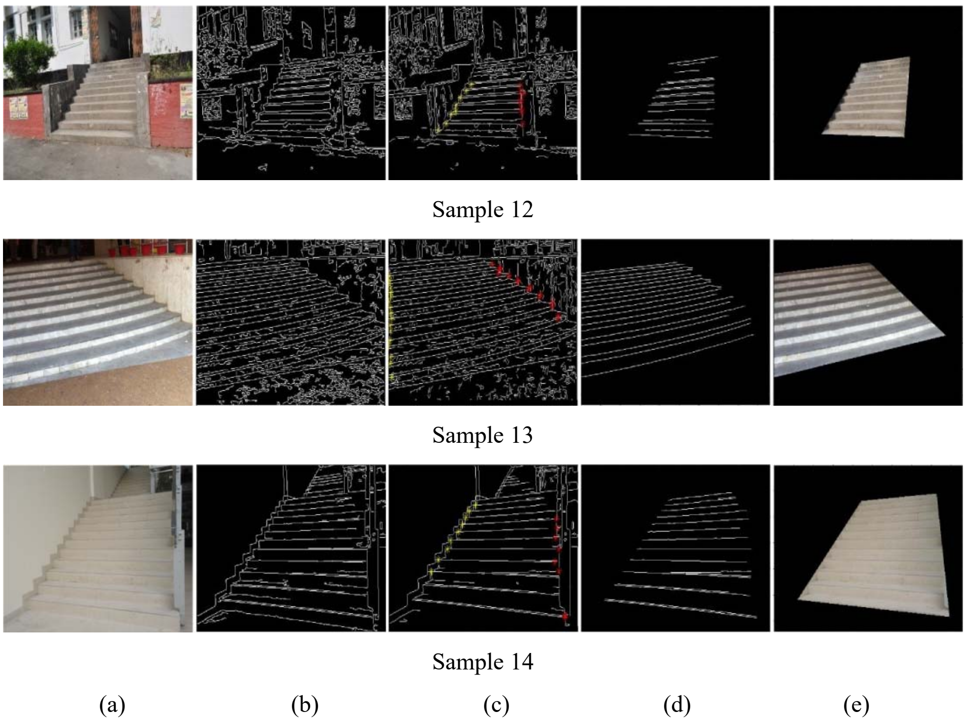


Figure 17 shows some processing example at different orientations. Where, stair sample image 12 and sample image 13 are captured from outdoor environments and sample image 14 is captured from indoor environments in different illumination condition. In the case of capturing different oriented images, some conditions should be considered that are the camera should be as close as possible from the horizontal axis and the angular displacement should not be more than 30 degrees. Experimental results in Figure 16 and Figure 17 show that, the proposed framework can detect the stair region from the given stair image efficiently if the framework detects the stair edges effectively.

The detection accuracy is calculated from various types of stair images which are taken from different environment and illumination conditions. The detection accuracy at different environmental conditions is as shown in Table 2. The experimental result shows the higher response to the indoor stairs with normal and uneven illumination conditions. It also gives a satisfactory response for the outdoor environmental conditions.

Table 2 Detection accuracy for different environment conditions

Stair type	*Environmental conditions*	*Total image*	*Correctly detected*	*Detection accuracy (%)*
Indoor	Normal and uneven illumination	42	41	97.62
Outdoor	Normal and uneven illumination	63	61	96.83
	Average	105	102	97.14

The proposed framework has been compared with Deb et al. (2013) and Hernández and Jo (2010) with respect to average detection accuracy and computation time shown in Table 3. Deb et al. (2013) used the dynamic programming to find out the longest increasing subsequence of stair horizontal edges. This problem is solved in O($n\ log\ n$) time. On the other hand, in this paper, the stair region is detected with the key feature of TCP. The TCP's are measured from the stair edge image within $O(n)$ time, which is linear, where; n is the number of horizontal parallel edges of a stair. Hernández and Jo (2010), Deb et al. (2013), and the proposed methods average detection accuracy and computation time are computed over 6, 100, and 105 images respectively.

Table 3 Comparisons with respect to stair detection accuracy and average computational time

Method	*Detection accuracy (%)*	*Avg. time (s)*
Hernández et al. (2010)	93.83	0.149
Deb et al. (2013)	96.15	0.085
The proposed method	97.14	0.067

Table 4 Distance estimation of sample images using proposed framework

Stair sample	*Actual distance from first camera (m)*	*Estimated distance (m)*	*Absolute error (m)*	*Accuracy (%)*	*Run time (s)*
Sample 3	4.5	4.46	0.04	99.11	0.073
Sample 4	6.25	6.07	0.18	97.12	0.072
Sample 5	11.75	11.38	0.37	96.85	0.075

The distance estimation accuracy of different stair sample images in Figure 14 is as shown in Table 4. The experimental result of additional distance estimation is shown in Table 5. The result reveals that the distance estimation is considerably accurate with an error rate less than 3.15%. Increasing distance from a camera to stair decreases the distance estimation accuracy. The reason is that increasing distance from a camera to stair decreases ($1-\alpha$) which is obvious from the description in (8).

Table 5 Distance estimation using proposed framework

SL no,	*Actual distance from first camera (m)*	*Estimated distance (m)*	*Absolute error (m)*	*Accuracy (%)*	*Run time (s)*
1	6.0	5.87	0.13	97.83	0.073
2	9.25	8.98	0.23	97.08	0.075

The proposed distance estimation framework has been compared with Hernández et al. (2011) with respect to distance estimation accuracy and average computational time is shown in Table 6.

Table 6 Comparisons with respect to distance estimation accuracy and average computational time

Method	*Distance estimation accuracy (%)*	*Avg. time (s)*
Hernández et al. (2011)	96.65	0.078
The proposed method	97.61	0.074

6 Conclusions

In this paper, a framework has been presented for extracting stair region from a stair image and estimating distance from camera to stair automatically without any prior information about the position of the stairways in a stair image. These steps are essential to improve the climbing process in an unknown environment for autonomous navigation system. For that, some unique and natural properties of a stair are utilised in this framework. This framework was tested successfully by a group of different stair images with varying styles and different illumination conditions. All the stair images are captured from the front side of stairs and with a height of less than 2 metres from the ground. The proposed framework detected stair regions of these stair images with acceptable running time and accuracy. The distance estimation from the camera to the stair is also satisfactory with an average accuracy of 97.61%. The experiments demonstrate the effectiveness of the proposed framework. The proposed framework is limited to the detection of traditional stairs with usual shapes. This framework may not give better response for stairs with unusual shapes such as spiral stair. In the future, this framework will be extended to detect the stair region from stair images with unusual shapes and implemented for detecting the stair region from downstairs image. And also focus will be given to improve the accuracy of distance estimation.

References

Barnard, S.T. (1983) 'Interpreting perspective images', *Artificial Intelligence*, Vol. 21, No. 4, pp.435–462.

Basca, C.A., Remus, B. and Lucian, B. (2007) 'Texture segmentation. Gabor filter bank optimization using genetic algorithms', *International Conference on Computer Tool*.

Canny, J. (1986) 'A computational approach to edge detection', *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 8, No. 6, pp.679–698.

Carbonara, S. and Guaragnella, C. (2014) 'Efficient stairs detection algorithm assisted navigation for vision impaired people', *2014 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA)*, June, pp.313–318.

Chen, L. (2012) 'Laplacian embedded regression for scalable manifold regularization', *Neural Networks and Learning Systems, IEEE Transactions*, June, Vol. 23, No. 6, pp.902–915.

Cong, Y., Li, X., Liu, L. and Tang, Y. (2008) 'A stairway detection algorithm based on vision for UGV stair climbing', *IEEE International Conference on Networking, Sensing and Control (ICNSC)*, April, IEEE, pp.1806–1811.

Daugman, J.G. (1985) 'Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two dimensional visual cortical filters', *J. Optical Soc. Amer.*, Vol. 2, No. 7, pp.1160–1169.

Deb, K., Islam, S.T., Sultana, K.Z. and Jo, K.H. (2013) 'Stairway detection based on extraction of longest increasing subsequence of horizontal edges and vanishing point', *Contemporary Challenges and Solutions in Applied Artificial Intelligence*, pp.213–218, Springer International Publishing, Switzerland.

Delmerico, J.A., Baran, D., David, P., Ryde, J. and Corso, J.J. (2013) 'Ascending stairway modeling from dense depth imagery for traversability analysis', *2013 IEEE International Conference on Robotics and Automation (ICRA)*, May, IEEE, pp.2283–2290.

duBuf, J., Barroso, J., Rodrigues, J., Paredes, H., Farrajota, M., Fernandes, H., José, J., Teixeira, V. and Saleiro, M. (2011) 'The smart vision navigation prototype for blind users', *JDCTA: Int. J. of Digital Content Technology and its Applications*, Vol. 5, No. 5, pp.351–361.

Duda, R. and Hart, P. (1972) 'Use of the Hough transform to detect lines and curves in pictures', *Comm. ACM*, Vol. 15, No. 1, pp.11–15.

Hernández, D.C. and Jo, K.H. (2010) 'Outdoor stairway segmentation using vertical vanishing point and directional filter', *2010 International Forum on Strategic Technology (IFOST)*, October, IEEE, pp.82–86.

Hernández, D.C., Kim, T. and Jo, K.H. (2011) 'Stairway detection based on single camera by motion stereo', *Modern Approaches in Applied Intelligence*, pp.338–347, Springer, Berlin, Heidelberg.

Huertas, A. and Medioni, G. (1986) 'Detection of intensity changes with sub pixel accuracy using Laplacian-Gaussian masks', *IEEE Trans. on Pattern Analysis and Machine Intelligence, PAMI*, Vol. 8, No. 5, pp.651–664.

José, J., Rodrigues, J. and du Buf, J. (2012) 'Visual navigation for the blind: path and obstacle detection', *International Conference on Pattern Recognition Applications and Methods*, Vol. 2, pp.515–519.

Lee, T.S. (1996) 'Image representation using 2D Gabor wavelets', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 10, pp.959–971.

Lee, Y., Leung, T. and Medioni, G. (2012) 'Real-time staircase detection from a wearable stereo system', *21st International Conference on Pattern Recognition (ICPR)*, November, Tsukuba, Japan, pp.3770–3773.

Lu, X. and Manduchi, R. (2005) 'Detection and localization of curbs and stairways using stereo vision', *International Conference on Robots and Automation*.

Maini, R. and Aggarwal, H. (2009) 'Study and comparison of various image edge detection techniques', *International Journal of Image Processing (IJIP)*, Vol. 3, No. 1, pp.1–11.

McLean, G.F. and Kotturi, D. (1995) 'Vanishing point detection by line clustering', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No.11, pp.1090–1095.

Schaffalitzky, F. and Zisserman, A. (2000) 'Planar grouping for automatic detection of vanishing lines and points', *Image and Vision Computing*, Vol. 18, No. 9, pp.647–658.

Schwarze, T. and Zhong, Z. (2015) 'Stair detection and tracking from egocentric stereo vision', *2015 IEEE International Conference on Image Processing (ICIP)*, September, IEEE, pp.2690–2694.

Se, S. and Brady, M. (2000) 'Vision-based detection of stair-cases', *Proceedings of Fourth Asian Conference on Computer Vision (ACCV)*, pp.535–540.

Shahrabadi, S., Rodrigues, J.M. and Du Buf, J.H. (2013) 'Detection of indoor and outdoor stairs', *Pattern Recognition and Image Analysis*, pp.847–854, Springer, Berlin, Heidelberg.

Shrivakshan, G.T. and Chandrasekar, C. (2012) 'A comparison of various edge detection techniques used in image processing,' *IJCSI International Journal of Computer Science Issues*, Vol. 9, No. 5, pp.272–276.

Wang, S., Pan, H., Zhang, C. and Tian, Y. (2014) 'RGB-D image-based detection of stairs, pedestrian crosswalks and traffic signs', *Journal of Visual Communication and Image Representation*, Vol. 25, No. 2, pp.263–272.

Weldon, T.P., Higgins, W.E. and Dunn, D.F. (1996) 'Efficient Gabor filter design for texture segmentation', *Pattern Recognition*, Vol. 29, No.12, pp.2005–2015.

Zhong, C., Zhuang, Y. and Wang, W. (2011) 'Stairway detection using Gabor filter and FFPG', *International Conference of Soft Computing and Pattern Recognition (SoCPaR)*, October, pp.578–582.