Collision-Free Navigation for Blind Persons using Stereo Matching

Saeid Fazli, Hajar Mohammadi D.

Abstract— A blind person in an urban environment has to navigate around obstacles and hazards. Though a significant amount of work has been done on detecting obstacles, not much attention has been given to the detection of drop offs, e.g., sidewalk curbs, downward stairs, and other hazards where an error could lead to disastrous consequences. In this paper, we propose algorithms for detecting obstacles in an urban setting using stereo vision and Two-Stage Dynamic Programming (TSDP) technique. We are developing computer vision algorithms for sensing important terrain features as an aid to blind navigation, which interpret visual information obtained from images collected by cameras mounted on camera legs nearly as high as young person. This paper focuses specifically on a novel computer vision algorithm for detecting obstacles, which are important and ubiquitous features on and near sidewalks and other walkways. A fast and robust stereo matching algorithm is proposed that uses only features in order to find a semi-dense disparity map. It works by growing from a small set of correspondence seeds. In this paper we use Normalized Cross-correlation (NCC) function matching with a 5 × 5 window and prepare an edge detected matching table τ and start growing disparity components by drawing a seed s from S which is computed using Sobel edge detector, and adding it to τ. It results in high accuracy and performance. The obstacle identified in the proposed method which appears in the disparity map enters the phase of depth computing. The algorithm extracts as much information as possible from depth information obtained from a stereo system. The proposed method is tested on a Pentium 4 Dual CPU 2.00 GHz PC with 2.00 GB of RAM and achieves better RMS error and bad matching percentage compared to previous methods. Experimental results are demonstrated on typical sidewalk scenes to show the effectiveness of the proposed method.

Index Terms— Obstacle detection, stereo matching, negative obstacle, positive obstacle, blind navigation, visually impaired.

---

1 INTRODUCTION

Obstacle detection is an important task for many mobile robots and visually impaired/blind assistance applications. Imagine a visually impaired/blind person is walking from home to market to shop for necessary needs, go to office, banks and etc., he/she would likely have to walk through corridors avoiding people and furniture (obstacles), along sidewalks at a safe distance from the curb, and possibly cross streets. Thus, for walking in his/her environment he/she has to be able to detect and avoid obstacles and drop-offs (and other hazards), i.e., safety is essential. Table 1 categorizes some common hazards that a visually impaired/blind person or robot can expect to find in an urban setting [1]. A good obstacle detection system must be capable of the following [2]:

a) To detect obstacles on a given space in good time

b) To detect and identify correct obstacles

c) To identify and ignore ground features that may appear as obstacles

TABLE 1: Common hazards in urban environments

Hazards	Examples
Drop offs	Sidewalk curbs, downward stairs, steps
Obstacles: Static Dynamic Invisible	Walls, furniture People, doors Glass doors and glass walls
Overhangs	Table tops, railings, tree branches
Inclines	Wheelchair ramps, curb cuts
Rough surfaces	Gravel paths, grass beds
Narrow regions	Doorways, elevators

• *Saeid Fazli is currently workingatZanjan university, Iran, PH-+98 9126420970. E-mail:fazli@znu.ac.ir*

This paper aims to research one of the computer vision's most important contributions to the navigation and obstacle detection for a visually impaired/blind person. Popular sensors for range-based obstacle detection systems include ultrasonic sensors [3], [4], laser rangefinders[5], radar[6] and stereo vision[7]. As these sensors measure the distances from obstacles to the person, they are inherently suited for the tasks of obstacle detection and avoidance. Laser range-finders have predominantly been the sensor of choice because they provide accurate range data with very little noise. Range sensors are also unable to distinguish between different types of ground surfaces. This is a problem especially in outdoors, where range sensors are usually unable to differentiate between the sidewalk pavement and adjacent flat grassy areas. Cameras are used less frequently, and as a secondary information source because they are sensitive to environmental changes and the data they return is more difficult to interpret [8]. But, the information from a camera can also be used for purposes other than safety, e.g., object recognition.

Stereo vision is an area of study in the field of machine vision that attempts to recreate the human vision system by using two or more 2D views of the same scene to derive 3D depth information of the scene. As an emerging technology, stereo vision algorithms are constantly being revised and developed. Stereo vision based obstacle detection is an algorithm that aims to detect and compute obstacle depth using stereo matching and disparity map. Depth information can be used to track moving objects in 3D space, gather distance information for scene features, or to construct a 3D spatial model of a scene.

Similar idea of using stereo images for the visually impaired/blind people‟ s navigation has been proposed in [9] and [10]. Stereo vision based methods are quite commonly used for obstacle detection and navigation. Gutmann et al. [11] proposed a stereo vision-based navigation system for humanoid robots, using a 2.5D grid to represent the world, each grid cell having a height and a label (floor/obstacle). To get accurate floor heights, the raw range data is segmented into planes. James Coughlan and

Huiying Shen[12] proposed a method to detect curbs based on stereo vision for blind wheelchair users but small changes in aligning of pair images incapacitate their method. If this system is mounted on a blind person‟ s head, the changes will be unavoidable. Two calibrated laser sensors and stereo camera are used by Aniket Murarka et al. [1] for color segmentation and motion based obstacle detection. Although this method is robust to detect both obstacles and drop-offs, it is quite expensive and complicated and still suffers from the changes in aligning of pair images.

Most works explained in the previous paragraph used stereo system with an active sensor but our research is just based on stereo vision and focuses on a different domain, in which the features of interest are difficult to detect from depth information alone, because the depth changes that characterize these obstacles may be small relative to the distances at which they are viewed and may be swamped by noise in the depth information estimated from stereo vision algorithms. Since an error can lead to disastrous consequences, it is essential for the visually impaired/blind people to detect native obstacles with a high degree of accuracy and hence we experimentally measure the error rates of the algorithms. We use cameras as the only sensors instead of lasers or other active sensors to simulate human vision system.

The next section presents some algorithms on stereo matching including segment-based stereo matching using belief propagation, disparity component growing and dynamic programming. Segment-based stereo matching using belief propagation is explained only for comparing to the proposed method, but two other algorithms (GCS and Dynamic programming) are also used in the proposed method. The proposed algorithm is explained in section III. It combines area based and feature based matching algorithms using growing component seeds and TSDP optimization. The obstacle detection phase is based on adaptive thresholding for positive obstacle and disentail of disparity map for negative obstacle. Section IV contains some experimental results on well-known databases and compares the results of the proposed method to the other ones. Some advantages of the proposed method are also discussed in this section. .Finally conclusion forms the last section.

2 MATCHING ALGORITHM

Matching techniques can be divided broadly into pixel-based, area-based and feature-based image matching, or a combination of them. Other types of stereo matching methods such as diffusion-based[13], wavelet-based[14], phase-based[13], and filter-based[15] have also been developed that are used less than the first three types i.e. pixel-based, area-based and feature-based methods.

The global algorithms formulate the problem of the disparity computation as an energy-minimizing problem and the smoothness assumption of the disparity map is usually made, there is some kind of function matching for stereo matching such as; Sum of Squared Difference (SSD), Sum of Absolute Difference (SAD), Sum of Hamming Distance (SHD) and Normalized Cross Correlation (NCC) [16],[17], [18] as follow:

$$NCC(f,g) = \frac{(f-\bar{f}).(g-\bar{g})}{\left|f-\bar{f}\right|.\left|g-\bar{g}\right|} \quad (1)$$

Let f be the intensity value of an M×N image and the left image within a box. We also have a similar definition for a second image g shifted d pixels in the x direction.

For a particular feature or a local window in one image, there are usually several matching candidates in the other image. It is usually necessary to use additional information or constraints to assist in obtaining the correct match. Some of the commonly used constraints are: (1) Epipolar constraint: Under this constraint, the matching points must lie on the corresponding epipolar lines of the two images. For epipolar rectified images, the matching points lie on the same image scanlines of a stereo pair; (2) Uniqueness constraint: Matching should be unique between the two images; (3) Smoothness constraint: Local regions of the disparity map should be relatively smooth apart from regions with occlusion or disparity discontinuity; and (4) Ordering constraint or monotonicity constraint: For points along the epipolar line in one image of the image pair, the corresponding points have to occur in the same order on the corresponding epipolar line in the other image. Other constraints mentioned will be used in the dynamic programming stage when obtaining the disparity map from the correlation coefficient volume [19].

1. Segment-Based Stereo Matching Using Belief Propagation

Belief Propagation is an iterative inference algorithm that propagates messages in the network [13]. Lankton[20] offered the segment-based stereo matching using belief propagation algorithm. The first step here is to get an estimate of the disparity at each pixel in the image. As the two images ‟ slide‟ over one another he subtracts their intensity values. Additionally, he subtracts gradient information from the images (spatial derivatives). Combining these give better accuracy, especially on surfaces with texture. He records the offset when the difference is the smallest as well as the value of the difference. He performs this slide-and-subtract operation from right-to-left (R-L) and left-to-right (L-R). Then we try to eliminate bad pixels in two ways. First, he uses the disparity from the R-L pass or the L-R pass depending on which has the lowest matching difference. Next, marks as bad all points where the R-L disparity is significantly different from the L-R disparity. In the next step, he combines image information with the pixel disparities to get a cleaner disparity map. First, he segments the reference image using a technique called “Mean Shift Segmentation.” This is a clustering algorithm that “over-segments” the image. The result is a very „blocky‟ version of the original image. Then he modified the algorithm by looking at the associated pixel disparities, for each segment.

In this paper we call the Lankton‟ s algorithm as“algorithm1” and the modified one as “algorithm2” in order to compare our proposed method in section IV.

2. Disparity component growth

The basic idea is to grow contiguous components in disparity space from initial correspondence seeds sorted in decreasing value of image similarity and to stop the growth process at image pixels where uniqueness constraint would be violated. The growth occurs in the neighborhood of previous matches in disparity space. This creates new seeds that are put to the priority queue.

Suppose we are given an unsorted list of disparity seeds S. Each seed is a point in disparity space, s = (x, x', y). Its neighborhood N(s) in disparity space consists of 16 point constructed from four sub-sets N1(s) ∪ N2(s) ∪ N3(s) ∪ N4(s), see Fig. 1 [21].

$$N1(s) = \{(x − 1, x' − 1, y), (x − 2, x' − 1, y), (x − 1, x' − 2, y)\} \quad (2)$$

$$N2(s) = \{(x + 1, x' + 1, y), (x + 2, x' + 1, y), (x + 1, x' + 2, y)\} \quad (3)$$

$$N3(s) = \{(x, x', y − 1), (x \pm 1, x', y − 1), (x, x' \pm 1, y − 1)\} \quad (4)$$

$$N4(s) = \{(x, x', y + 1), (x \pm 1, x', y + 1), (x, x' \pm 1, y + 1)\} \quad (5)$$

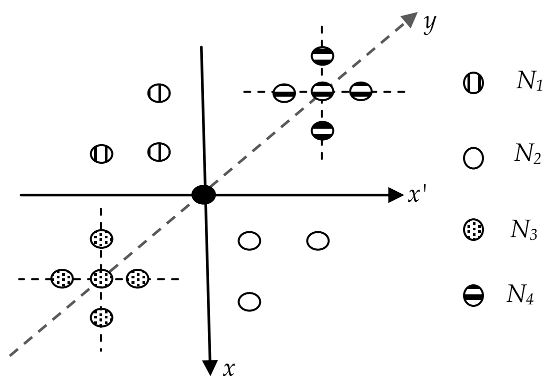


Figure 1. Disparity space neighborhood used in this paper.

The neighborhood is selected so as to limit the magnitude of disparity gradient to unity and to improve the ability to follow a disparity component even if the image similarity peak falls in between pixels in the matching table. Assuming similarity is computed from small image windows around pixels (u, v) and (u', v) by the normalized cross-correlation, we prepare an empty matching table τ and start growing disparity components by drawing an arbitrary seed s from S, adding it to τ, individually selecting the best-similarity neighbors qi over its four sub-neighborhoods Ni(s):

$$qi = (u, u', v)i = \underbrace{\arg\max}_{(x, x', y) \in Ni(s)} simil(x, x', y) \quad (6)$$

and adding these neighbors qi to the seed list if their interimage similarity exceeds a threshold. Hence, up to four new seeds are created. If we draw a seed from the list S that is already a member of the matching table, then we discard it. The growth must stop in a finite number of steps by exhausting the list S. The output from the growth phase is a partially filled matching table whose connected regions in 3D represent disparity components grown around the initial seeds. Chec et al use the maximum Strict Sub-Kernel (SSK) for matching. For more information about SSK you can see [22], [23],[24].

3. Dynamic programming

Dynamic programming strategy has a cost matrix with the nodes representing the weight of matching a pixel in the left image with a pixel in the right image. The cost of matching pixel x in the left image and pixel y in the right image can be computed based on the costs of matching all pixels in the left of these two pixels. If one assumes the ordering constraint, the optimal path computed to match the pixels in left and right images will result in the best set of matches for the pixels in left and right images.

Dynamic programming is an effective strategy to compute correspondences for pixels. This method is finding the minimum cost path going monotonically down and right from the top-left corner of the graph to its bottom-right corner [25]. Different optimization techniques, including dynamic programming (DP)[26], graph cuts [27], [7], and genetic algorithm[28], have been applied under this framework. These techniques try to find a global optimized solution under some given parameters. However, due to the complex nature of the stereo vision problem, it is difficult, if not impossible, to have a universal set of parameters that can produce good disparity maps for different stereo images. In fact, even within a single image, it is highly possible that different regions should use different parameters since the signal-tow noise ratio may vary within the image. As a result, the best solution in terms of the given parameters may not necessarily be a good solution [29], [30].

The key difference between dynamic programming and simple recursion is: a dynamic programming algorithm memorizes the solutions of optimal sub-problems in an organized, tabular form (a dynamic programming matrix), so that each sub-problem is solved just once.

Dynamic programming matches lines to lines. They can also use the matches found for previous pixels in the same scan line in the computation for the subsequent pixels[31], [32]. In[33], Sun used the matching problem as an optimization one of the total cross correlation scores over a surface through a 3D cross correlation volume (height, width, and the disparity range of a region) and the matching is solved efficiently using a two-stage dynamic programming algorithm, where cross correlation scores are maximized along paths in the 3D cross correlation volume. Note that by using a global matching criterion, accurate cross correlations are needed at all locations since they affect the optimal path estimation and thus the matching. As cross correlations can change significantly even at sub-pixel level, in order to achieve optimal matching, sub-pixel accuracy cross correlations need to be calculated [34].

3 PROPOSED METHOD

In this work we focus on detecting positive/negative obstacles in outdoor environments using vision-based algorithms optimized by dynamic programming. We use cameras as the primary sensors instead of lasers, RADARs, ultrasounds, etc because cameras significantly provide much more information in a single frame of data. The proposed method consists of three phases including computing disparity map, detecting positive/negative obstacles and computing depth. For positive obstacle detection, matching phase is the second phase and for negative obstacle detection, is the first one. Detecting positive obstacle doesn" t need disparity map thus at first, we detect the positive obstacles then disparity map is computed. In this way, the complexity of processing decreases because only the disparity of obstacles is computed not the whole image. As negative obstacle detection is based on differential of disparity map therefore we have to apply the matching phase and compute disparity map to the whole image. The contributions of the proposed method are proposing a novel feature-area based stereo matching algorithm and also obstacle detection using adaptive threshoding and adaptive mean filter.

In the proposed method, the matching phase is divided into two stages; primary matching which is area based and secondary matching which is feature based. Then on every feature of the second stage, we use the above-mentioned window of special neighbors. Although we achieve a robust matching, but optimization of the disparity map using TSDP is applied to decrease errors. Dense depth map calculation is a process, in which, given two images Il and Ir, each pixel of Il is matched to a pixel in Ir. In our case the match can be expressed as a displacement for each pixel p(x; y) ∈ Il to a pixel q(x′; y′) ∈ Ir. The problem of selecting optimal matches between pixels is solved by DP. The DP approach returns an optimal match for each scan line of the images, but the differences between scan lines is a typical problem of this method. Thus, DP yields the optimal path through grid. This is the best set of matches that satisfy the ordering constraint.

1. Positive obstacle detection

In this paper positive obstacles are categorized into staircases and others. This method is easy to implement and fast enough for obstacle avoidance in highly textured environments. Then adaptive thresholding, resizing and noise removal are performed. The adaptive thresholding is based on the average intensity of the whole image. For staircases the detection region must contain at least 3 concave and convex edges alternately. The proposed method determines the number of stairs as follows:

$$\text{Number of stairs} = NPE + 1 \tag{7}$$

NPE is the number of convex and concave edge pairs. The last edge is sigle concave edge so NPE must be increased one unit.

Then disparity map and depth are computed. This method is useful and accurate for obstacle avoidance. Removing the texture and the obtrusive details causes decreasing complexity in computing disparity map. This method is easy to implement and fast enough for obstacle avoidance.

2. Negative obstacle detection

It is necessary to know that negative obstacles are more significant to detect because the risk of such obstacles are more than others. Since an error can lead to disastrous consequences, it is essential for a blind person to detect negative obstacles with a high degree of accuracy and hence we experimentally measure the error rates of the algorithm. The proposed algorithm is a robust method in detecting negative obstacles using growing correspondence seeds on detected edges. A fast stereo matching technique is used to achieve near real time speed. Other calculations consume short time so they don‟t significantly affect the system speed. We combined MATLAB and C++ codes in this work.

As explained before, the global algorithms for matching formulate the problem of the disparity computation as an energy-minimizing problem and the smoothness assumption of the disparity map is usually made. The function matching in this paper is NCC function with demanded accuracy which is calculated by equation 4.

NCC function matching is used on a 5 × 5 window as image similarity statistic in all experimental results. We prepare an edge matching table τ and start growing disparity components by drawing a seed s from S which is computed using canny edge detector, adding it to τ . In this way we have higher performance than previous works such as [21]. In this paper, Jan Cech and Radim Sara use random seeds but we use matrix S which contains edges of the left and the right images. In this way, we combine area based and feature based matching, thus the accuracy is increased for real images. Another advantage of the proposed method is applying edge feature matching on the neighborhood thus the calibration stage in our previous works [35],[36] is omitted. Matching edges, two stage matching and optimizing disparity map using dynamic programming guide us to achieve good performance even if calibration is omitted. As the motion of a blind person‟s head is unavoidable while navigating, the alignment of stereo images might be unreliable in the stereo system.

A fast stereo matching algorithm is proposed that visits only a small fraction of disparity space in order to find a semi-dense disparity map. It works by growing from a small set of correspondence seeds. Unlike in known seed-growing algorithms, it guarantees matching accuracy and correctness, even in the presence of repetitive patterns. The quality of correspondence seeds influences computing time, not the quality of the final disparity map.

For optimizing disparity map, we use two-stage dynamic programming. After some iterations of DP, accuracy is increased compared to the one stage DP.

Finally depth of obstacle is computed as follows:

$$Z(x,y) = \frac{f.B}{d(x,y)} \tag{8}$$

f is focal length, B is base line (distance of camera centers), d(x,y) is disparity value in x row and y column.

Pseudo-code of the proposed method is shown in TABLE 2.

4 EXPERIMENTAL RESULTS

Here, we present some experimental results to show strength of the proposed method.

1. Example 1 from CMU CLI database

The image “Bowling2” from CMU CLI database is chosen as the first example because it contains many similar segments and therefore requires more efficient matching algorithm. Fig.2 shows the original image and the ground truth of the “Bowling2”. Outputs of the four mentioned algorithms including “algorithm1”, “algorithm2” (the modified version of “algorithm1”), GCS and the proposed method are presented in fig. 3. Although there are many similar segments in the images, as can be seen the proposed method achieves higher similarity to the ground truth.

ISSN 2229-5518

Table 2 Pseudo-code of the proposed method.

Negative obstacle detection
1. calculating elementary disparity map with low accuracy
2. calculating minimum and maximum value of the disparity map
3. feature extraction by canny edge detector (feature: edge)
4. calculating disparity based GCS algorithm in 3D boxes for every feature
5. optimizing disparity map by TSDP
6. applying adaptive mean filter (first stage of removing noise)
7. canny edge detection with threshold 0.07
8. second stage of removing noise
9. determining the negative obstacles' position
10. calculating the depth of negative obstacles
Positive obstacle detection
1. pre-processing (enhancement of images)
2. adaptive thresholding
3. calculating elementary disparity map with low accuracy
4. calculating minimum and maximum value of the disparity map
5. feature extraction by canny edge detector (feature: edge)
6. calculating disparity based GCS algorithm in 3D boxes for every feature
7. optimizing disparity map by TSDP
8. determining the positive obstacles' position
9. calculating the depth of positive obstacles

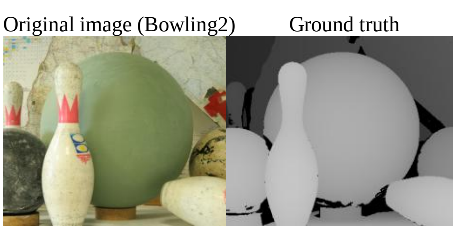


Figure 2. The original image and ground truth of Bowling2

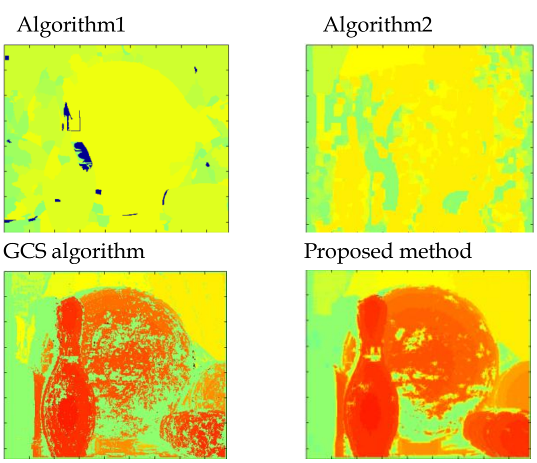


Figure 3. From left to right, disparity map of "algorithm1", "algorithm2" (the modified of "algorithm1"), GCS algorithm and the proposed method, respectively

2. Example 2 (a positive obstacle)

Fig.4 shows the original images. In these two images not only the images are not calibrated, but also the scene is changed. The results of the four algorithms are presented in fig. 5. Although there are many differences in the left and right images; the proposed method achieves more accurate disparity map than other methods.

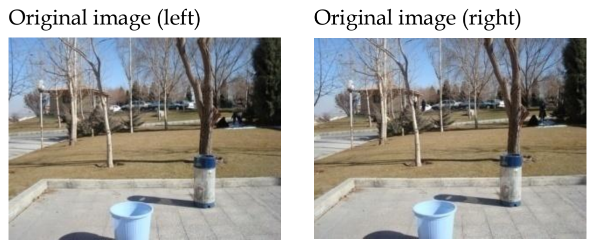


Figure 4. The original image

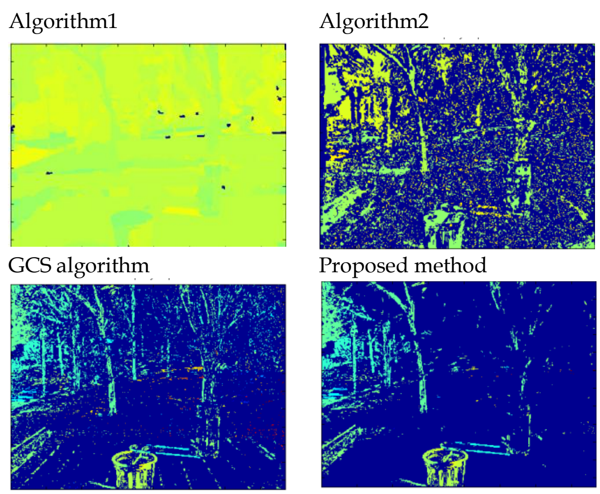


Figure 5. From left to right, disparity map of "algorithm1", "algorithm2" (the modified of "algorithm1"), GCS algorithm and the proposed method, respectively

3. Example 3 from database [12]

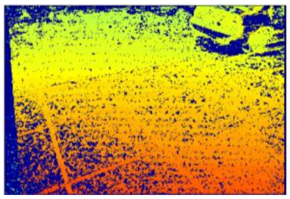
The database contains some rectified images of a road curb photographed from a 12 foot distance. The focal length of the camera and the distance between cameras were 498.11 pixels and 12cm, respectively. Fig.6 shows the example of obstacles and all stages of phase 2 of the proposed method. The negative obstacle (curb) is detected and the distance of obstacle (depth) is calculated correctly as shown later. Fig.7 shows the original images and every stage of both phases 2 and 3 of the proposed method. The detected negative obstacle is shown in the red rectangle. The depth of the negative obstacle is calculated using Eq. 8 which is 3.1 meters in this case.

Obstacle detection using only GCS algorithm for the same scene of fig. 6 is presented in fig.7. Comparison of fig. 6 and fig. 7 reveals that the proposed algorithm detects the right negative edge even in the presence of some changes in the texture of ground as can be seen in the original image. GCS algorithm results in incorrect negative edges as shown in fig. 7.

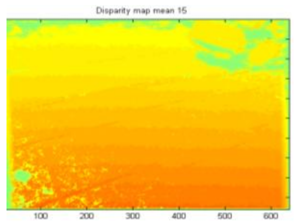
Original left image

Original right image

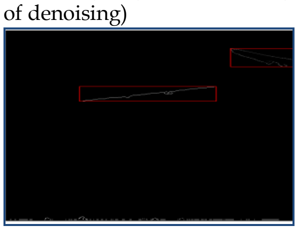
Disparity map (proposed method)



Adaptive mean filter for occlusion pixel (first stage of denoising)



Removing edges (second stage of denoising)



Detected obstacle

Figure 6. The example of negative obstacles and output of the proposed method

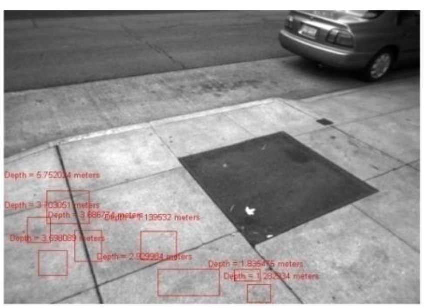


Figure 7. negative obstacle detection using GCS algorithm

4. Example of detecting both positive and negative obstacles using the proposed algorithm

Results of applying the proposed algorithm to simultaneously detect positive and negative obstacles are shown in fig.8 and 9.

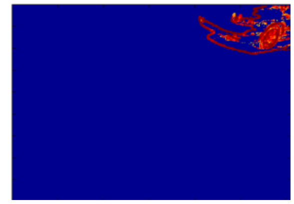
5. Quantitative evaluation of the methods

In this section we calculate some quantitative criteria to compare the mentioned algorithms. Well-known images including "Flowerpots", "Baby3", "Aloe", "Bowling2", "cones", "poster", "Larch" and "Head" were chosen. Table 3 shows the run time of the matching stage for the four algorithms on the mentioned images. The results show that the proposed method achieves higher speed than other methods.

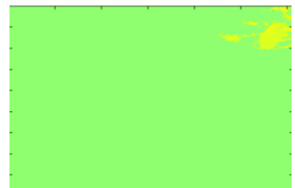
The matching time shown in table 3 is for MATLAB code running on a Pentium 4 Dual CPU 2.00 GHz PC with 2.00 GB of RAM.

Daptive thresholding by thresh=0.3

Correlation of left and right images



Disparity map



Obstacle detected and its depth

Figure 8. The output of the proposed method of positive obstacles step by step

Figure 9. detecting both positive and negative obstacles

TABLE 3, Time (sec.) consuming for 5 shown images from various datasets

Algorithms / Name	Algorithm1	Algorithm2	GCS	Proposed method
Flowerpots	8.2517	9.3562	5.20313	4.85938
Baby3	9.2846	9.1283	4.0625	2.73438
Aloe	10.7995	8.8941	3.39013	2.29688
Larch	48.5997	27.0116	29	4.96875
Head	67.9420	59.2115	34.8281	9.28125

Table 4 presents RMS errors of the methods using;

$$RMSError = \sqrt{[\sum (D(i,j) - G(i,j))^2]/n} \quad (11)$$

The table reveals that the proposed algorithm results in better RMS error scores compared to "algorithm1" and GCS but for some images "algorithm2" works better.

TABLE 4, RMS Errors for the 5 Images from various Datasets

Algorithms / Name	Algorithm1	Algorithm2	GCS	Proposed method
Bowling2	300.2140	270.3512	233.2678	225.5857
Flowerpots	412.365	116.265	351.0103	203.4086
Baby3	201.354	116.425	193.145	193.1745
Aloe	150.321	96.265	125.4934	109.2895
Cones	212.142	163.251	147.185	129.232
poster	100.32	93	91.94	66.90

The last two tables and figures 3 and 5 show that the results of GCS and the proposed method are more similar than others thus percentages of bad matching are computed just for these two algorithms and shown in table 5. The proposed method demonstrates better results than the GCS algorithm.

TABLE 5, Percentages of Bad Matching

Algorithms / Name	GCS	Proposed method
poster	5.97	4.47
cones	6.3	5
Aloe	6.53	5. 3
Baby3	10.86	9.29
Flowerpots	5.57	5.53
Bowling2	10.7	6.85

5 Conclusion

In this paper we proposed a novel method to detect the positive/negative obstacles for visually impaired/blind people. The proposed method contains three phases and two stages of matching: elementary matching which is area based and secondary matching which is feature based. In this way we benefit from the advantages of both area and feature based algorithms. After two stage of matching, the disparity map is optimized using TSDP. The system is just based on stereo cameras and no active sensor is used so it is inexpensive and also gives more information about an environment for a visually impaired/blind person. The experimental results are presented from well known datasets for various scenes including a ground with many edges inside and some scenes with changes in the texture. The proposed method achieves better accuracy and speed (minimum run time) than other existing methods. The results shown in the last section proves that the proposed method is quite fast and robust for both negative and positive obstacle detection.

References

[1] A. Murarka and e. al, "Detecting Obstacles and Drop-offs using Stereo and Motion Cues for Safe Local Motion," in IEEE/RSJ International conference on Intelligent Robots and Systems, 2008, pp. 702-708.

[2] S. Rahman, "Obstacle Detection for Mobile Robots Using Computer Vision," Department of Computer Science University of York Final Year Project, 2005.

[3] O. Ringdahl, "Techniques and Algorithms for Autonomous Vehicles in Forest Environment," Department of

Computing Science Umeå University Licentiate Thesis 978-91-7264-373-4, 2007.

[4] W. Samakming and J. Srinonchat, "Development Image Processing Technique for Climbing Stair of Small Humanoid Robot," in International Conference on Computer Science and Information Technology, 2008, pp. 616-619.

[5] C. Y. Sung, T. S. Jin, and J. M. Lee, "Optimal Moving Windows for Real-Time Road Image Processing," Journal of Robotic Systems 20(2), 65–77 (2003), vol. 20, no. 2, pp. 65-77, 2003.

[6] A. Angelova, L. Matthies, D. Helmick, and P. Perona, "Learning and Prediction of Slip from Visual Information," Journal of Field Robotics, vol. 24, no. 3, p. 205–231, 2007.

[7] P. Foggia, ,. J.-M. Jolion, A. Limongiello, and M. Vento, "Stereo Vision for Obstacle Detection: a Graph-Based Approach," in 6th IAPR – TC-15 Workshop on Graph-based Representations in Pattern Recognition, 2007, pp. 11-13.

[8] S. Thrun and e. al, "Stanley: The Robot that Won the DARPA Grand Challenge," Journal of Field Robotics, 2006.

[9] S. Meers and K. Ward, "A vision system for providing 3D perception of the environment via transcutaneous electro-neural stimulation," in The 8th International Conference on Information, 2004, pp. 546-552.

[10] G. balakrishnan, G. Sainarayanan, R. Nagarajan, and S. Yaccob, "On stereo processing procedure applied towards blind navigation aid-SVETA," in The 8th International Symposium on Signal Processing and Its Applications, 2005, pp. 567-570.

[11] J. Gutmann, M. Fukuchi, and M. Fujita, "Real-time path planning for humanoid robot navigation," in IJCAI, 2005.

[12] j. Coughlan and H. Shen, "TERRAIN ANALYSIS FOR BLIND WHEELCHAIR USERS: COMPUTER VISION ALGORITHMS FOR FINDING CURBS AND OTHER NEGATIVE OBSTACLES," in Conference & Workshop on Assistive Technologies for People with Vision & Hearing Impairments Assistive Technology for All Ages CVHI, 2007.

[13] D. Sharstein, r. Szeliski, and R. Zabih, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," in IJCV, 2002, p. 7–42.

[14] p. Bellutta, r. Manduchi, l. Matthies, K. Owens, and A. Rankin, "Terrain Perception for DEMO III," in Intelligent Vehicle Symposium, 2000.

[15] P. Moallem and K. Faez, "Effective Parameters in Search Space Reduction Used in a Fast Edge-Based Stereo Matching," Journal of Circuits, Systems, and Computers, vol. 14(2), pp. 249-266, 2005.

[16] A. Stein and M. Hebert, "Local detection of occlusion boundaries in video," in BMVC, 2006.

[17] S. B. Pollard, J. E. W. Mayhew, and J. P. Frisby, "PMF: A stereo correspondence algorithm using a disparity gradient constraint. Perception," Perception, vol. 14, pp. 449-470, 1985.

[18] R. M. Perception and L. G. Shapiro, "Image segmentation techniques," CVGIP, vol. 29, pp. 100-132, 1985.

[19] G. balakrishnan, G. Sainarayanan, R. Nagarajan, and S. Yaccob, "On stereo processing procedure applied towards blind navigation aid-SVETA," in The 8th International Symposium on Signal Processing and Its Applications, 2005, pp. 567-570.

[20] S. Lankton, "Sparse Field Methods - Technical Report," Georgia Institute of Technology, 2009.

[21] J. Cech and R. Sara, "Efficient Sampling of Disparity Space for Fast and Accurate Matching," in BenCOS Workshop CVPR, 2007.

[22] R. Sara, "Finding the largest unambiguous component of stereo matching.," in ECCV, 2002, pp. 900-914.

[23] S. Radim, "Robust Correspondence Recognition for Computer Vision.," in 17th Conference of IASC-ERS , Roma, 2006.

[24] D. Gusfield and R. W. Irving, "The Stable Marriage Problem: Structure and Algorithms," in The MIT Press,, 1989.

[25] H. Mohammadi D., S. Shirazi Tehrani, and P. Moallem, "A Novel Obstacle Detection Method using Stereo Vision and two-stage Dynamic Programming," in IEEE/ICEE, 2010.

[26] D. Scharstein and R. Szeliski, "Stereo matching with nonlinear diffusion," International Journal of Computer Vision," vol. 28, p. 155–174, 1998.

[27] D. Jones and J. Malik, "Computational framework for determining stereo correspondence from a set of linear spatial filters," image processing, vol. 76, p. 321–331, 2000.

[28] Y. S. Kim, J. Lee, and Y. Ha, "Stereo matching algorithm based on modified wavelet decomposition process," Pattern Recognition,, vol. 30, p. 929–952, 1997.

[29] j. Chul Kim and e. al, "A Dense Stereo Matching Using Two-Pass Dynamic Programming with Generalized Ground Control Points," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, p. 1075–1082.

[30] M. Gong and Y. H. Yang, "Fast Stereo Matching Using Reliability-Based Dynamic Programming and Consistency Constraints," in Ninth IEEE International Conference on Computer Vision , 2003.

[31] S. Birchfield and C. Tomasi, "Depth discontinuities by pixel-to-pixel stereo",," International Journal of Computer Vision,, p. 269–293, 1999.

[32] B. Tang, D. Ait-Boudaoud, B. J. Matuszewski, and L. k. Shark, "An Efficient Feature Based Matching Algorithm for Stereo Images," in Proceedings of the Geometric Modeling and Imaging-New Trends (GMAI‟ 06), 2006, pp. 195-202.

[33] C. Sun, "Fast Algorithms for Stereo Matching and Motion Estimation," in in Proceedings of Australia-Japan Advanced Workshop on Computer Vision,, Adelaide, 2003, pp. 38-48.

[34] A. Donate, Y. Wang, X. Liu, and E. Collins, "Efficient and Accurate Subpixel Path Based Stereo Matching," in 19th International Conference on Pattern Recognition, IEEE, 2008.

[35] ,. S. Fazli, H. Mohammadi D, and P. Moallem, "Obstacle Detection using Sum of Haming Distance," in ICEEE, 2009, pp. 23-30.

[36] S. Fazli, H. Mohammadi D, and P. Moallem, "A Robust Obstacle Detection Method in Highly Textured Environments using Stereo Vision," in 2nd international conference on machine vision (IEEE),, 2009, pp. 97-100.