韓國컴퓨터情報學會 論文誌
第16卷 第12號, 2011. 12.

2011-16-12-2-1

탑-뷰 변환과 빔-레이 모델을 이용한 영상기반 보행 안내 시스템

림 청*, 한 영 준**, 한 헌 수***

Vision-based Walking Guidance System Using Top-view Transform and Beam-ray Model

Qing Lin*, Young-Joon Han**, Hern-Soo Hahn***

요 약

본 논문은 야외 환경에서 하나의 카메라를 이용한 시각 장애인을 위한 보행 안내 시스템을 제안한다. 기존의 스테레오 비전을 이용한 보행 지원 시스템과는 다르게 제안된 시스템은 사용자의 허리에 고정된 하나의 카메라를 이용하여 꼭 필요한 정보만을 얻는 것을 목표로 하는 시스템이다. 제안하는 시스템은 먼저 탑-뷰 영상을 생성하고, 생성된 탑-뷰 영상 내 지역적인 코너 극점을 검출한다. 검출된 극점에서 방사형의 히스토그램을 분석하여 장애물을 검출한다. 그리고 사용자 움직임은 사용자에 가까운 지역 안에서 옵티컬 플로우를 사용하여 추정한다. 이렇게 영상으로부터 추출된 정보들을 기반으로 음성 메시지 생성 모듈은 보행 지시 정보를 합성된 음성을 통해 시각 장애인에게 전달한다. 다양한 실험 영상들을 사용하여 제안한 보행 안내 시스템이 일반 인도에서 유용한 안내 지시를 제공하는 것이 가능함을 보인다.

▸ Keyword : 보행 안내 시스템, 탑-뷰 변환, 옵티컬 프로우, 멀티모달 정보 변환

Abstract

This paper presents a walking guidance system for blind pedestrians in an outdoor environment using just one single camera. Unlike many existing travel-aid systems that rely on stereo-vision, the proposed system aims to get necessary information of the road environment by using just single camera fixed at the belly of the user. To achieve this goal, a top-view image of the road is used, on which obstacles are detected by first extracting local extreme points and then verified by the

• 제1저자 : 림청 • 교신저자 : 한헌수
• 투고일 : 2011. 10. 05, 심사일 : 2011. 10. 10, 게재확정일 : 2011. 10. 19.
* 숭실대학교 전자공학과(Dept. of Electronic Engineering, Soongsil University)
** 숭실대학교 전자공학과(Dept. of Electronic Engineering, Soongsil University)
*** 숭실대학교 전자공학과(Dept. of Electronic Engineering, Soongsil University)

polar edge histogram. Meanwhile, user motion is estimated by using optical flow in an area close to the user. Based on these information extracted from image domain, an audio message generation scheme is proposed to deliver guidance instructions via synthetic voice to the blind user. Experiments with several sidewalk video-clips show that the proposed walking guidance system is able to provide useful guidance instructions under certain sidewalk environments.

▸ Keyword : walking guidance system, top-view transform, optical flow, multimodal information transform

I. Introduction

Authoritative statistics have shown that about 1% of the world population is visually impaired, and among them about 10% is fully blind. One of the consequences of being visually impaired is the limitations in mobility. Therefore, many electronic travel-aids devices have been developed to provide assistance to sight-impaired people in a certain local environment.

Electronic travel-aids system can be categorized depending on how the information is gathered from the environment and how to deliver to the user[1]. For example, information can be gathered with ultrasonic sensor, laser scanners, or cameras, and user can be informed via auditory[2][3][4][5] or tactile device[6][7][8]. In recent years, vision-based travel-aids systems using camera have won more attentions due to advantages like larger sensing area, higher sensing resolution as well as low cost.

Most existing vision-based travel-aids systems have been developed using stereo vision methods. In these systems, stereo cameras are used to create a depth map of the surrounding environment, and then this depth map is transformed into stereo sound or tactile devices for the use of self-navigation by visually impaired people.

For instance, Mora [5] developed a navigation device to transform depth map into a kind of stereo sound space, and delivered to the user via headphones. While the TVS system developed by Johnson and Higgins[6] transformed depth information into a tactile belt with 14 vibration motors laterally attached. Also the Tyflos navigator system[7] converts depth map into vibration sensing on a 2-D vibration array that is attached on the user's abdomen. The element of the array that vibrates represents the direction, where an object is detected and the different vibration levels represent the distance of the object. ENVS system developed by Meers[8] transformed depth map to electrical pulses that stimulate the nerves in the skin via electrodes located in the TENS data gloves.

Although many stereo-vision based travel-aids systems have been proved to be effective under certain environment, there are still some problems existed. First of all, due to the computation load of stereo vision algorithm, most systems just simply run stereo vision algorithm and directly convey the depth information to the user without doing any further obstacle detection and avoidance algorithm. As a consequence, users have to do obstacle avoidance themselves by sensing and judging the transformed auditory or tactile pattern from the depth map, which makes the system less easy to use and requires much user training. Secondly, user's self-walking motion is seldom considered in the existing stereo-vision based system, which is obviously very helpful for the autonomous navigation.

To handle these problems existed in the stereo-vision based travel-aid system, a walking guidance system using just one single camera is proposed in this paper. The general flowchart of the system is shown in Fig.1

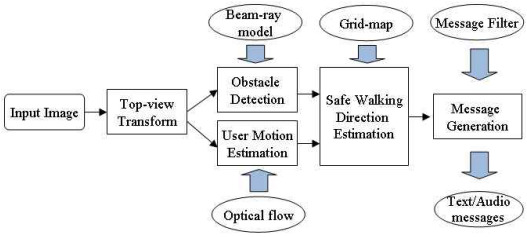

Fig. 1. General flowchart of the system.

II. Top-view Transform

The first step of the whole algorithm is to map the input image onto a virtual plane which is located at the top of the road plane. This top-view transform will bring several advantages.

First of all, the bottom point of erect obstacle is much easier to detect on top-view, which is very important for obstacle localization in single camera metrology. Secondly, obstacle edges are stretched while the road surface edges in the near field are suppressed. With more obvious bottom points and stretched edge components, it would be easier to detect obstacle position on the top-view. Finally, since perspective effect is removed on top-view image, the moving speed of image pixels with respect to the user's motion is more uniformly distributed from bottom to the top of image, which makes the user motion estimation much easier.

Inverse perspective mapping (IPM) is used to generate a top-view from the input image. The knowledge of the camera parameters is required for the application of the IPM transform. The camera model for top-view transform is illustrated in Fig.2.

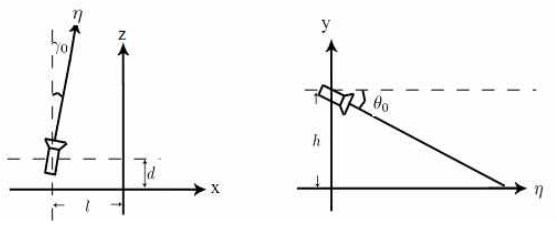

Fig. 2. Camera model for top-view transform.

Mathematically, IPM can be modeled as a projection from a 3D space W, containing elements $(x, y, z) \in R^3$, onto a 2D planar subspace I, with elements $(u, v) \in R^2$. The transformation from W to I is given as in equation (1).

$$u(x,0,z) = \frac{[\arctan\{\frac{h\sin\gamma(x,0,z)}{z-d}\} - (\theta_0 - \alpha_u)] \cdot (m-1)}{2\alpha_u}$$
$$v(x,0,z) = \frac{[\arctan\{\frac{z-d}{x-l}\} - (\gamma_0 - \alpha_v)] \cdot (n-1)}{2\alpha_v} \quad \cdots\cdots (1)$$

The important parameters in (1) are as follows:

① Camera Viewing point is defined by camera position in world coordinate system C={l, h, d}.

② Camera Viewing direction is described by two angles r and Θ, as shown in Fig.2.

③ Camera angular aperture is 2αυ in row direction and 2αv in column direction. αu = arctan(μ0 / F), αv= arctan(v0 / F), (μ0, v0) is the camera's focal center, and F is focal length.

④ Camera resolution is n × m.

By using (1) and filling the corresponding camera parameters, the transform between perspective view and top-view domains can be easily carried out. Fig.3 shows an example of top-view transform using IPM.

(a) Perspective-view image (b) Top-view image

Fig. 3. An example of top-view transform.

III. Obstacle Detection

1. Beam-ray model

In sidewalk environment, the most common obstacles are thin and erect obstacles like trees, poles and other pedestrians. These obstacles appear in a kind of beam-ray pattern on top-view image. As is shown in Fig.4, on top-view image, this beam-ray

pattern is composed of two parts: the bottom point and two edge components ejecting from the bottom point. Therefore, the detection of obstacles is composed of two steps. The first step is to detect candidate bottom points, and then the edge components associated with each candidate bottom points are checked.

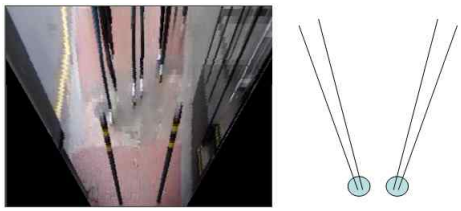

Fig. 4. Beam-ray model.

2. Candidate bottom points detection

It can be observed that the bottom points usually have different gray level values with its neighborhood pixels on the top-view image. To model this kind of feature, scale-invariant feature transform (SIFT) is generally used on the perspective view domain, like that used in [9]. However, in terms of top-view domain, the scale and rotation variation is not obvious. Therefore, to reduce the computation load of SIFT, these bottom points are just modeled as local maxima or minima points of its neighborhood in terms of gray level values on the top-view image.

To extract these local extreme points, we group the pixels in a 5 by 5 window, and then fit a second order polynomial using the 25 points in this 5 by 5 neighborhood.

$$f(x,y) = p_1 + p_2 x + p_3 y + p_4 x^2 + p_5 y^2 + p_6 xy \quad (2)$$

As is shown in (2), second order polynomial function f(x,y) include six coefficients. Once the 6 coefficients of this polynomial function have been determined, the partial derivatives of this polynomial function f(x, y) can be checked using Hessian matrix H(f(x,y):

$$H(f(x,y)) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix} \quad \cdots\cdots (3$$

The eigen-values of Hessian matrix correspond to the curvature of the gray value surface. If both eigen-values are negative, the point is a local maximum, and if both are positive, it's a local minimum. Also, if they have different signs, it's a saddle point. Therefore, a point is accepted to be a local extreme point if the absolute values of both eigen-values of the Hessian matrix are greater than predefined. Fig.5 shows an example of local extreme points extracted on top-view image, where local extreme points are labeled in red dots.

3. Edge component verification

After these local extreme points are extracted, the edge components associated with each candidate bottom points are checked for beam-ray model verification.

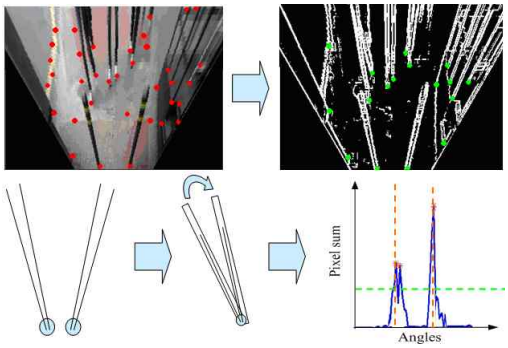

Fig. 5. Obstacle detection based on beam-ray model.

As is shown in Fig.5, at each candidate bottom point, we sample one direction and check the number of edge pixels that lie on this direction. As a result, for each direction angle, the number of edge pixels that lie on this direction can be obtained. By accumulating all the sampled directions, a polar histogram can be calculated for each candidate bottom point. The horizontal axis in the histogram is

direction angles and vertical axis is number of edge pixels. If two distinct peaks are detected on the histogram, and the peak angle difference lie in a reasonable range, then this bottom point is accepted as real obstacle bottom point, otherwise the points are discarded.

IV. Safe Walking Direction Estimation

1. User motion estimation

Assuming that user is located at the bottom center of the top-view image, and a neighboring area close to the user's position is supposed to be composed of mainly road surface pixels. Therefore, we select a rectangular area around center bottom position for evaluating the optical-flow vector field as shown in Fig.6 (a).

Considering that the walking speed of user is not fast, and the ROI is very close to the user, there will be very little illumination variations for the corresponding pixels in consecutive images of an image sequence. In this way, constancy of the gray values, which is a very important assumption in optical-flow, can be ensured. In addition, in this walking scenario, there will not be big displacements. Based on these considerations, combined local-global method proposed by Bruhn et al.[10] is used for calculating the optical-flow vector field.

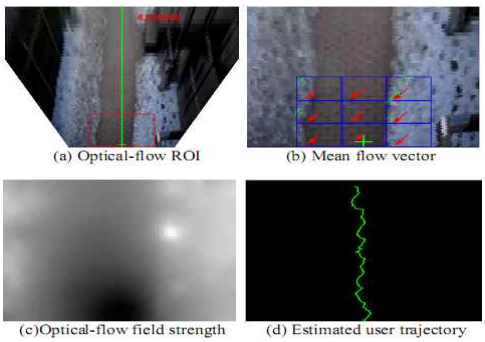

Fig. 6. User motion estimation using optical-flow.

For each pixel location in this rectangle area, a motion vector can be obtained, with its magnitude represents moving speed and its direction represents moving direction. The estimated optical-flow strength inside the ROI region is shown in Fig.6(c). In order to estimate user's walking speed and walking direction more robustly, the ROI region is divided into 9 sub-regions, and the mean motion vector is calculated in every sub-region, then by comparing the differences between those 9 mean vectors, we group the similar ones into clusters, and use the mean vectors included in the largest cluster to calculate final mean vectors. This helps to exclude outlier motion vector to eliminate their influence to the final mean motion vector. This final mean vector is then used to represent user's walking speed and walking direction. Based on user's walking speed and direction in each frame, the user walking trajectory in a certain time period can be generated, as is shown in Fig.6(d).

2. Safe walking direction estimation

By knowing the obstacle positions as well as user's walking motion, the next step is to estimate safe walking direction based on a top-view grid map. The basic idea is to generate a top-view grid map for each possible moving direction that we defined. And then on the grid map, for each defined moving direction, a direction score is calculated, while the direction with the highest score is selected as the suggested safe moving direction for the next step.

Grid map is defined by n×n grids. The value of n is equivalent to the average human step length. The position of user in the graph is assumed to be located at the middle bottom of the image plane, while the obstacle locations are labeled using small circles on the graph. An example of top-view grid map is shown in Fig.7.

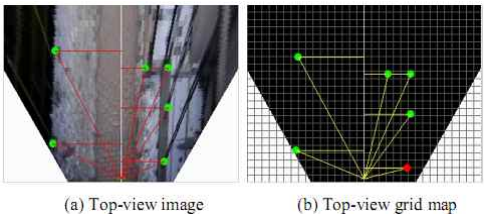

Fig. 7. Top-view grid map generation.

On the grid map, five moving directions are defined, as is shown in Fig.8. Direction 0 indicates straight, direction 1 for left 45°, direction 2 for hard left, direction 3 for right 45°, and direction 4 for hard right. Based on the knowledge of user's motion in the scene, a grid map can be generated for each possible moving direction after a certain period of time. Fig.8 shows an example of grid map generation, where three grid maps are generated for three moving direction respectively.

Firstly, the grid map for 0 direction is generated, and direction score is calculated. If the direction score for 0 direction is larger than a warning threshold, then the user is suggested to keep the straight moving direction. Otherwise, the grid map for other moving directions will be generated and direction scores are calculated. And the direction with highest score is chosen as the next suggested moving direction. However, if the largest direction score is still less than a predefined "stop" threshold, then "stop" instruction is generated instead.

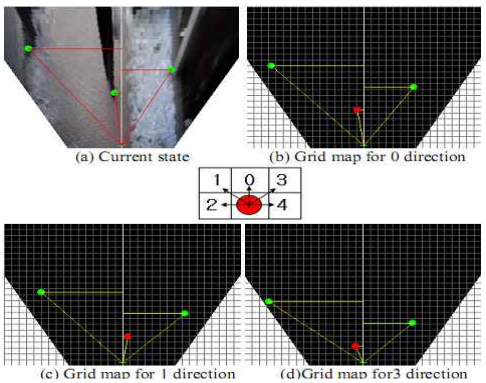

Fig.8. Grid map for possible moving directions.

The direction score is calculated by considering the smallest triangular area formed by obstacle, user and user's walking direction. As shown in Fig.8, for each obstacle, a triangular can be formed. The smaller triangular area is, the more dangerous this obstacle will be; therefore, this smallest triangular area can be used as a direction score for evaluating the safety of the current walking direction. On the grid maps shown in Fig.8, the obstacle with smallest triangular area is considered as the most dangerous obstacle and is labeled in red color.

IV. Message Generation

By applying the above monocular vision algorithm s on the top-view image, three types of necessary information for guidance can be obtained. As is shown in Fig.9, these three types of information include safe walking direction, obstacle positions as well as user's walking motion. The next important step is to transform the information obtained from the image domain to the language domain, and deliver the verbal messages to the user in an appropriate manner.

As is shown in Fig.9, in order to achieve the information transform between two domains, three types of sentence patterns are defined for each type of information obtained from image domain, and the parameter values in real numbers are mapped to predefined template words.

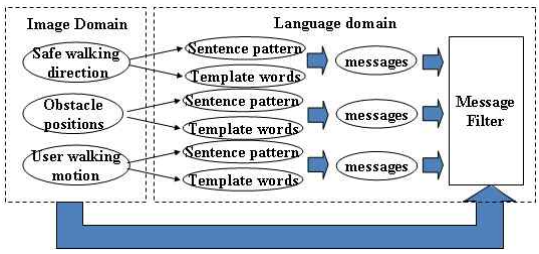

Fig. 9. Information transform from image to text domain.

In addition, a message filter is proposed which uses different combination of rules to select the most suitable messages that should be delivered to the

user at this moment. An example for message generation is shown in Fig.10. Three types of information are transformed to the language domain, including safe walking direction: ①, obstacle information:②,③,⑤,⑥,⑦, and safe walking speed: ④.

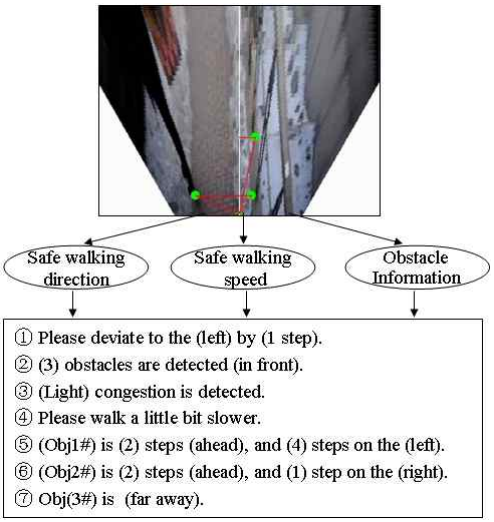

Fig. 10. An example of message generation

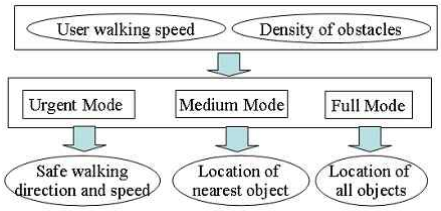

Fig. 11. Message filter mode definition.

In order to deliver only the most appropriate set of messages to the user, a message filter is designed to filter messages in three types of mode. As illustrated in Fig.11, in urgent mode, only messages regarding safe walking direction and speed are delivered, in medium mode, the message of nearest object is also delivered, while in full mode, the messages of all the detected objects are delivered.

The message filter mode is determined by user walking speed S, and the density of obstacles D. A mode score W is estimated using S and D via (4), where a represents the factor weights that S or D is considered. The larger this score is, the more urgent the situation would be.

$$W = S \bullet \partial + D \bullet (1 - \partial) \quad \cdots\cdots (4)$$

V. Experimental Results

The whole software part of the walking guidance system is developed using C++ under windows platform. To test the performance of our system, we attached a camera on a belt and fix it at user's belly, looking a little bit downward to the road ahead of user. The camera captures images of the road environment and processed by the system software which runs at a lap-top computer carrying in user's backpack. The generated messages are turned into synthetic voice and delivered to the user via a loudspeaker.

The system is tested in different sidewalk environment around our campus. Fig. 11 shows an example of the testing results. In this scene, the original input image is shown in Fig.12(a), the detected obstacle bottom points are shown in Fig.12(b), while user motion is estimated by optical-flow field shown in Fig.12(c). Then based on the results of Fig.12(b) and Fig.12(c), grid map for the 0 direction is generated, as in Fig.12(d) shows, and step score is calculated. The step score in this scene is higher than the alarming threshold, therefore, user is supposed to move straight forward. The recommended moving direction is indicated by red arrow shown in Fig.12(a).

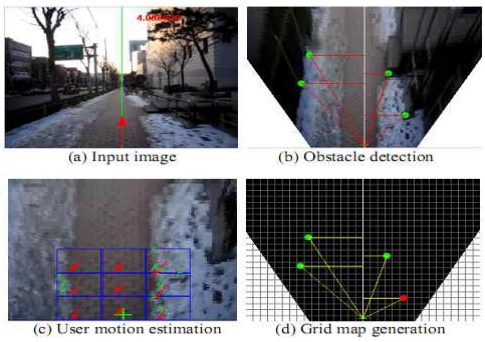

Fig. 12. Test results example 1#.

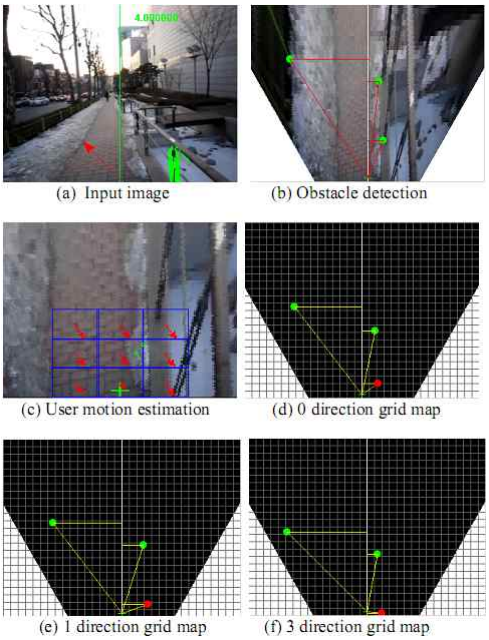

Fig. 13. Test results example 2#.

Another scene of experimental result is shown in Fig.13. In this scene, the grid map for straight direction is first generated, and the smallest triangular area is calculated. However, the smallest triangular area is less than the predefined warning level, therefore grid map for other moving directions are generated, and step scores for these moving directions are calculated, as is shown in Fig.13(e)(f). Since the left 45 degree moving direction has the largest step score on grid map, therefore, the suggested moving direction for next step is left 45 degree.

Fig.14 shows an example of the system output in medium mode (a), and in urgent mode (b). In Fig.14, the left image in 1st row shows original image, with red arrow displays the suggested safe walking direction, the right image in 1st row shows detected obstacle position on top-view image, the right image in 2nd row shows mean optical flow vector, while the left image in 2nd row displays generated messages under different working modes.

In medium mode, there are only 4 obstacles detected far away from the user, and estimated user walking speed is slow, in this case, user will be in a safe state in a relatively long time period, so the system is able to deliver more information to the user, like nearest obstacle information, safe walking direction information as well as suggestions for user's walking speed.

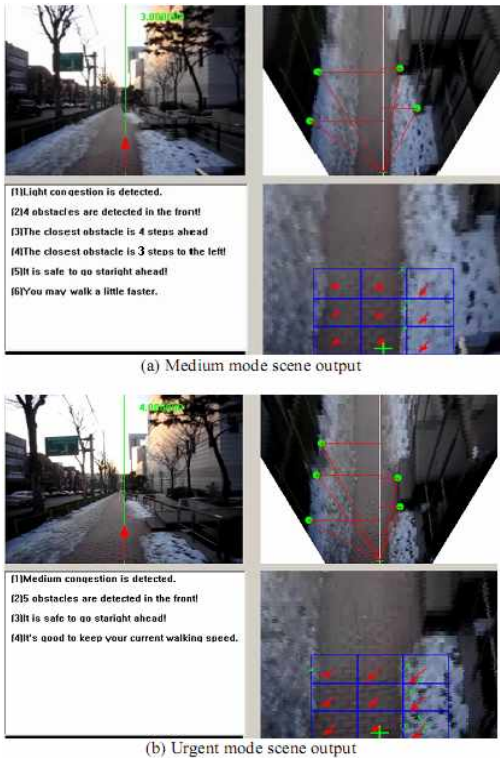

Fig. 14. Text generation results.

While under urgent mode case shown in Fig.14(b), there are more obstacles closer to the user are detected, and user's walking speed is fast, so in this case, only the most critical information like safe walking direction is delivered to the user.

The performance of the algorithm is evaluated in two parts: obstacle detection accuracy and safe walking direction accuracy. To evaluate the obstacle detection accuracy, we test the algorithm on a challenging side-walk video clip, with complex road side structures and cluttered road surface. Some sample images of the scene are shown in Fig.12 to Fig.14.

2000 frames from this video clip are used for testing. All the critical obstacle positions are manually labeled on the top-view images of these 2000 frames. If the obstacle is detected within a 5 pixels deviation from the ground truth position, then it is accepted as correct detection, otherwise it is counted as false detection. If the obstacle is not detected, it is counted as miss-detection. If some road surface clutters are detected as obstacles, it is regarded as false detection. The obstacle detection result is shown in Table 1.

In real testing, the Hessian matrix threshold in bottom point detection step is tuned to be a little lower to reduce the miss-detection cases, since miss-detection may cause serious mistakes in the following navigation steps. Therefore, miss-detection case happens only when obstacle bottom part look very similar to the road surface in terms of gray level values.

Table 1. Obstacle detection rate

Total Obstacles	Correct Detection	False Detection	Miss Detection
9486	7788 (82.1%)	1660 (17.5%)	38 (0.4%)

To evaluate the safe walking direction accuracy, we generate a virtual user walking trajectory by using the estimated safe walking direction and estimated user's self walking speed. And this virtual walking trajectory is then mapped to the top-view grid map with all the obstacles positions labeled. A segment of this synthesized map is shown in Fig.15.

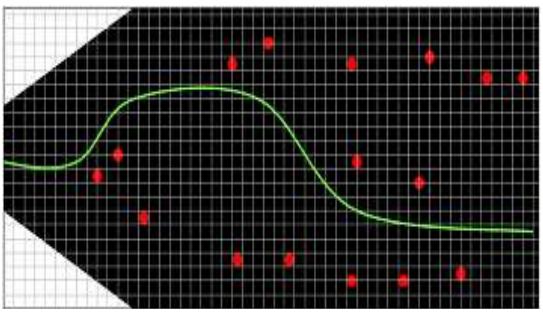

Fig.15 Synthesized walking trajectory and grid map.

VI. Conclusion

A walking guidance system for blind people in outdoor environment is proposed in this paper. Compared with general stereo vision based method, the proposed system handles this problem using just single camera. It not only can detect various erect obstacles with simple beam-ray model on top-view image, but also takes user motion into conside ration, which can provide more helpful guidance information for user. The future work would involve improving the human-machine interface for providin g guidance instructions to blind user in a more comfortable and friendly manner.

Acknowledgment

This work was supported by the Brain Korea 21 Project in 2011, and by the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency)(NIPA-2011-(C1090-1121- 0010))

References

[1] Dimitrios Dakopoulos and Nikolaos G. Bourbakis, "Wearable Obstacle Avoidance Electronic Travel Aids for Blind: A Surveyk," IEEE Transactions On Systems, Man, And Cybernetics, Vol. 40, No.1, pp. 25 - 35, January 2010.

[2] G. Sainarayanan, R. Nagarajan, and S. Yaacob, "Fuzzy Image Processing Scheme for Autonomous Navigation of Human Blind," Appl. Softw. Comput., Vol. 7, No. 1, pp. 257 - 264, January 2007.

[3] S. Y. Shin, H. Y. Moon, S. B. Pyo, "Image Recognition Using Colored-hear Transformation Based On Human Synesthesia ," Journal of The

Korean Society of Computer and Information, Vol.13, No.2, pp.135-141, March 2008.

[4] A. Hub, J. Diepstraten, and T. Ertl, “Design and Development of An Indoor Navigation and Object Identification System For The Blind,” in Proc. ACM SIGACCESS Accessibility Computing, No. 77-78, pp. 147-152, January 2004.

[5] J. L. G.-Mora, A. R.-Hern´ andez, L. F. R.-Ramos, L. D´ ıaz-Saco, and N. Sosa, "Development of A New Space Perception System for Blind People Based on The Creation of A Virtual Acoustic Space," http://www.iac.es/proyect/eavi.

[6] L. A. Johnson and C. M. Higgins, “A Navigation Aid For The Blind Using Tactile-visual Sensory Substitution,” in Proc. 28th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc., NewYork, pp.6298-6292, 2006.

[7] D. Dakopoulos, S. K. Boddhu, and N.Bourbakis, "A 2D Vibration Array Assistive Device For Visually Impaired," in Proc. 7th IEEE Int. Conf. Bioinf. Bioeng. ,Boston, MA, Vol.1, pp.930-937, 2007.

[8] S. Meers and K. Ward, "A Substitute Vision System for Providing 3D Perception and GPS Navigation via Electro-tactile Stimulation," in Proc. Int. Conf. Sens. Technol., New Zealand, November, 2005.

[9] Yi Hu, B. J. Shin, C. W. Lee, " Place Modeling and Recognition using Distribution of Scale Invariant Features," Journal of The Korean Society of Computer and Information, Vol.13, No.4, pp.51-58, July 2008.

[10] Bruhn, Weickert, Feddern, Kohlberger, and Schnö rr, "Lucas/Kanade Meets Horn/Schunck: Combini ng Local and Global Optic Flow Methods," Interna tional Journal of Computer Vision, Vol.61, No.3, 211-231, 2005.

저 자 소 개

림 청

2003 : 산동과학기술대 컴퓨터학과 공학사
2004 : 산동과학기술대 컴퓨터학과 강사
2010 : 숭실대학교 전자공학과 석사
현 재 : 숭실대학교 전자공학과 박사 과정
관심분야 : 영상처리, 물건검출, 임베디드 시스템
Email : lqsdust@163.com

한 영 준

1996 : 숭실대학교 전자공학과 학사
1998 : 숭실대학교 전자공학과 석사
2003 : 숭실대학교 전자공학과 박사
현 재 : 숭실대학교 전자공학과 부교수
관심분야 : 로봇비전, 영상처리, 비주얼서보잉
Email : young@ssu.ac.kr

한 헌 수

1981 : 숭실대학교 전자공학과 학사
1983 : 연세대학교 전자공학과 석사.
1986 : University of Southern California 석사
1991 : University of Southern California 박사
현 재 : 숭실대학교 전자공학과 교수
관심분야 : 자동화시스템, 자료융합, 물체인식
Email : hahn@ssu.ac.kr