

Using Computer Vision to See

Bogdan Mocanu^{1,2}, Ruxandra Tapu^{1,2}(✉), and Titus Zaharia²

¹ Telecommunication Department, Faculty of ETTI,
University “Politehnica” of Bucharest, Bucharest, Romania

² ARTEMIS Department, Institut Mines-Telecom/Telecom SudParis,
UMR CNRS MAP5 8145, Evry, France

{bogdan.mocanu,ruxandra.tapu,titus.zaharia}@telecom-sudparis.eu

Abstract. In this paper we propose a navigation assistant for visually impaired people, which uses computer vision techniques and is integrated on a wearable device. The system makes it possible to detect and recognize, in real-time, both static and dynamic objects existent in outdoor urban scenes without any a priori knowledge about the obstruction type or location. The detection system is based on relevant interest point extraction and tracking, background/camera motion estimation and foreground object identification through motion vectors clustering. The classification method receives as input image patches extracted by the detection module, performs global image representation using binary VLAD and prediction based on SVM. The feedback of our system is transmitted to visually impaired users through bone-conduction headphones as a set of audio warning messages. The entire system is fully integrated on a regular smartphone. The experimental evaluation performed on a set of 20 videos acquired with the help of VI users, demonstrates the pertinence of the proposed methodology.

Keywords: Assistive wearable device · Obstacle localization and recognition · Acoustic feedback · Visually impaired users

1 Introduction

For people suffering of visual impairment, common daily activities such as the autonomous navigation to a desired destination, familiar face recognition or independent buying of specific products can represent an important challenge. The safety displacement in outdoor scenario is very difficult because of VI people reduced capacity to understand and perceive the environment, the continuous change of the scene [1] or possible collision with moving objects (e.g. pedestrians, cars, bicycles or animals) or static obstructions (e.g. traffic signs, waste containers, fences, trees, etc.). If for a common setting the position of static hazards can be learned, the location estimation of dynamic obstacles is particularly difficult.

In an unknown setting most VI users relay on assistive devices such as the white canes or guiding dogs to acquire additional information about the potential obstructions. The white cane is effective in detecting objects situated directly in

front of the person and it requires an actual physical contact with obstruction. However, even though the white cane is largely accessible to anyone, it shows quickly its limitations when confronted with real life situations (i.e. it cannot identify further away or overhanging objects, it cannot offer additional information about the type of obstruction and its degree of danger)[2]. Even though the trained dogs help reducing some of the above shortcoming they are highly expensive, have reduce operational time and require an extensive training phase.

In this context, the present paper introduces a complete navigation assistance system that facilitates the safe displacement of visually impaired (VI) people in urban areas. The proposed solution aims at improving the life quality of VI users by increasing their mobility and willingness to travel. At the hardware level our solution is based on a regular smartphone device, bone conduction headphones and chest mounted harness. The core of our framework is represented by the smartphone used both as an acquisition system (i.e. the video camera and the gyroscope sensor) and as a processing unit. The proposed technology is low-cost because it does not require any dedicated hardware architecture but regular components accessible on the market. The modules are lightweight making the systems wearable and portable, satisfying the hands-free and ears-free requirements imposed by VI users. At the software level the major contribution of the paper is the introduction of a method, based on computer vision and machine learning techniques that works in real time, returning warning messages fast enough so that the user could walk normally. The algorithms were carefully designed and optimized in order to work efficiently on a low processing unit.

The rest of the paper is structured as follows. In Sect. 2 we review the state of the art in the domain of assistive technologies. The focus is put on wearable devices that use computer vision techniques. In Sect. 3 we describe in details the proposed obstacles recognition methodology (Sect. 3.1. obstacle detection and Sect. 3.2. object classification). Section 4 presents the experimental evaluation of the proposed framework. For testing we used actual VI users in real life situations with: various moving objects, irregular camera displacement, abrupt changes in the illumination conditions. In Sect. 5 we conclude our work and open some perspectives of future work.

2 Related Work

In the last couple of years various navigation assistance systems were introduced, designed to create a digital enhancement to the white cane. In this chapter we briefly describe and analyze the technical literature focusing on the main strengths and limitation of each framework.

One of the first methods introduced in the state of the art [3] offers obstacle detection and guided navigation functionalities by using commercial available hardware components. The system is difficult to carry, is invasive and cannot identify overhead obstacles. In [4], in order to differentiate between foreground and background obstructions a fuzzy neural network is employed. In [5], a CCD camera system that transforms the information from an obstacle detection module to a voice message system is proposed. An indoor and outdoor navigation

assistant called SmartVision is proposed in [6]. The system is highly depended on the quality of the GPS (Global Positioning System) signal acquired and on the initial position estimation.

In [7], the authors propose to mount the video camera on the VI user waist. The method is highly sensitive on the camera position and has never been tested on real life scenarios. Recently, with the development of the smartphone industry various authors proposed transforming the assistant into an Android application. In [8], the indoor obstacles, situated at arbitrary heights, are identified with high confidence scores. However, the system violates the hand-free conditions imposed by the VI user [9] and is not able to differentiate between obstacles. The authors of [10] introduce a novel obstacle recognition method that performs both detection and classification.

In [11], a navigation assistant based on depth map estimation is proposed. The system is designed to detect obstacles situated at arbitrary levels of heights. A wearable device that facilitates the safe indoor displacement is presented in [12]. The information about possible obstructions is stored as a metric map that is very difficult to exploit by VI users. In order to develop an object detection and localization method, the authors in [13] use 3D object reconstruction, while in [14], an overhead obstacle detection system is introduced based on 3D map and motion estimation using 6DOF. Recently, in [15], an indoor and outdoor navigation system has been proposed based on conditional random fields and depth maps.

The KinDetect system introduced in [16] is designed as an obstacle detection method that combines information coming from Kinect and depth sensors. A different method able to recognize various types of obstacles using a Kinect sensor is proposed in [17]. In [18], nearby structure information is converted into acoustic maps. Both systems [17] and [18] can perform the detection and recognition in real-time by using a powerful backpack processing unit. Recently in [19] an object detection and tracking together with a 3D map construction is introduced, while the authors in [20] develop a 3D face recognition method designed to help VI users identify familiar faces.

After analyzing the state of the art we can conclude that every method has its own advantages and limitations over the others, but no one can offer, in a satisfactory degree, all necessary features for the autonomous navigation of VI users in an unknown setting. Under this perspective, the proposed solution is designed not to replace the cane but to complete it with additional functionalities (*e.g.* guidance information, obstacle recognition capabilities and object degree of danger estimation).

The main contributions of the present paper concern: (1) an obstacle detection system based on relevant interest points extraction and tracking, camera motion estimation using multiple homographies per frame. By analyzing the motion information we estimate the location of various objects and their degree of danger relative to the VI user. (2) an obstacle recognition method using relevant interest points, global image representation using VLAD and SVM classification using one versus all strategy. (3) an acoustic feedback system that uses bone conduction headphones and stereo principles.

All the methods were designed and tuned to achieve real-time capabilities on light processing units as smartphone devices.

3 Proposed Approach

The autonomous navigation in outdoor environments can be facilitated by a system able to recognize obstructions (i.e. static and dynamic) and transmit alert messages. Our framework is designed to detect and semantically understand, in real time, potential dangerous situation and to warn VI users about various obstructions encountered along the walking path.

3.1 Obstacle Detection and Localization

Relevant Interest Point Extraction. The method starts by extracting interest points using the pyramidal FAST algorithm [21]. We have empirically observed that regular FAST method returns a high number of interest points even for low resolution images/videos. Moreover, the descriptor is focused on highly textured regions situated in most of the cases in the background while for foreground objects little or no information is extracted. Because our application is designed to work in real-time on a low processing device we decided to privilege a semi-dense sampling approach that reduces the total number of interest points. We overlap a regular grid over the first frame and we determine for each cell the associated interest points. Then, we propose to retain for each cell only one point, the most relevant, i.e. the one with the highest value of the Harris-Laplacian operator [22]. Different from other filtering strategies [23] that retain the best top-k points based on their magnitude value without considering any information about the spatial location our method insures that the points are better distributed within the image. Our strategy is able to capture more informational content of the image, while avoiding accumulation of points in certain textured areas. Figure 1 illustrates three different extraction strategies: Fig. 1a presents the interest points retained using the traditional FAST method; Fig. 1b shows the set of points obtained after applying the method introduced in [23], while Fig. 1c presents the results obtained with the above-described strategy. Let us underline that the examples illustrated in Fig. 1b and c contain the same number of interest points.

The performance of the approach and also the computational burden is controlled by the grid size defined as: $G_{size} = (W * H) / N_{points}$, where W and H are the width and height of the image, N_{points} is the maximum number of interest points we retain (e.g. for a video stream with the resolution of 320×240 pixels we fixed N_{points} to 1000 points).

Interest Point Tracking. The selected interest points are tracked between successive frames using the multiscale Lucas-Kanade (LK) algorithm [24]. Even though the LK tracking method proves to be sensitive to light variation and is inconsistent in estimating the motion vectors, we adopted LK algorithm

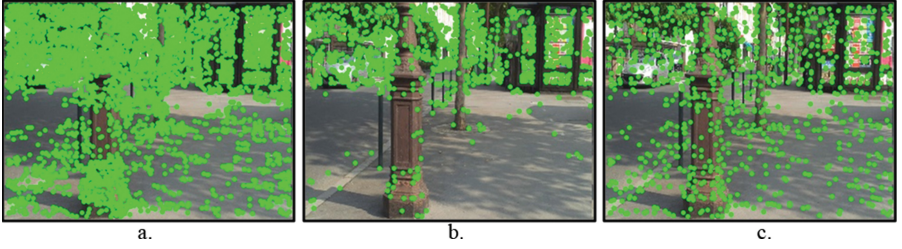


Fig. 1. Relevant interest point extraction using: (a) the traditional FAST method; (b) the strategy introduced in [23]; (c) our method based on regular grid filtering

because it offers the best compromise between the quality of the motion flow and the processing time. The LK method is initialized with the set of interest points extracted using the regular grid strategy. Then, the points are tracked within the video stream. However, for low textured zones within the image or for regions depicting objects disappearing or that reenter the scene, it is necessary to reinitialize the tracker. In such specific areas we locally apply as input to the LK algorithm the relevant set of points obtained (*cf.* grid strategy described in Sect. 3.1). If we denote one interest point within the reference image as: $p_{1m}(x_{1m}, y_{1m})$ then the correspondent one in the successive image, established using the tracking algorithm is denoted by $p_{2m}(x_{2m}, y_{2m})$. The associated motion vector $v_m(v_{mx}, v_{my})$ magnitude and orientation can be determined as:

$$D_{12} = \sqrt{v_{mx}^2, v_{my}^2} \quad ; \quad \theta_{12} = \text{arctg} \frac{v_{my}}{v_{mx}} + k\pi \quad (1)$$

Camera Motion Estimation. For the global motion model we have adopted the RANSAC algorithm [25] that estimates the optimal homographic matrix (H) between successive frames. For any point expressed in homogenous coordinates $p_{1m}(x_{1m}, y_{1m}, 1)^T$ we can determine its novel position ($p_{2m}^{est}(x_{2m}, y_{2m}, 1)^T$) in the adjacent frame by multiplying p_{1m} with H . Then, we can determine the prediction error (Er) by computing the Euclidian distance ($L_2norm \|\cdot\|$) between the current point position (establish using the tracking algorithm) and its predicted location:

$$Er(p_{2m}) = \|p_{2m}^{est} - p_{2m}\| \quad (2)$$

In the ideal case the prediction error is equal to zero. However, for real-life application we need to compare Er to a pre-established threshold (Th_1) in order to determine the set of inliers/outlier. The interest points belonging to the inliers class satisfy the transformation and belong to camera or background motion. The outlier class contains all other types of motion present in the scene.

Foreground Object Detection. We focus next in identifying different classes of motion existent in the scene. Due to the foreground apparent motion even static object (situated in the foreground) act like moving objects relatively to the global background displacement. A clustering analysis is performed on the outlier set of interest points, by taking into account both the motion vector magnitude and orientation. However, a direct clustering within the polar coordinates domain is not feasible. Typically, the motion vectors angle have a circular range between 0 and 2π , so a clustering algorithm based on the L_2 distance, such as k -means, which assume that the input data is distributed in the Cartesian space is inappropriate (since 0 and 2π should be interpreted as equivalent values). We propose performing a non-linear transformation from the polar coordinates to the 2D Cartesian domain using the following trigonometric relations:

$$v_{mx}^* = r \times \cos \theta_{12} \quad ; \quad v_{my}^* = r \times \sin \theta_{12} \quad (3)$$

where r represents the radial coordinate, that incorporates the magnitude information. The value of r is computed as: $r = 1 + D_{12}/D_{Max}$, with D_{Max} the maximum displacement of a motion vector. In this way we impose all points to lay on an annulus with the two radiuses equal to 1 and 2, respectively. Moreover, diametrically opposite points will not be cluster together. In Fig. 2 we give a graphical representation of the interest points distribution in the Cartesian domain by using motion vectors magnitude and orientations.

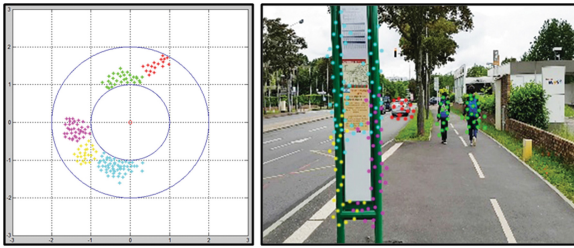


Fig. 2. Relevant interest point annular representation using motion vectors magnitude and orientations

The motion classes are determined after applying the k -means clustering algorithm [26], within the considered representation space:

$$\arg \min_{\{\alpha_1, \dots, \alpha_k\}} = \sum_{i=1}^k \sum_{v_m^* \in \Xi} \|v_m^* - \alpha_i\|^2 \quad (4)$$

where v_m^* is the novel motion vector $v_m^*(v_{mx}^*, v_{my}^*)$ associated to an interest point, α_i is the cluster centroid by averaging all interest points included in a class Ξ , k is the total number of clusters in k -means. The value of k is set to 5 in our experiments. Finally, we verify the clusters consistency by analyzing the interest

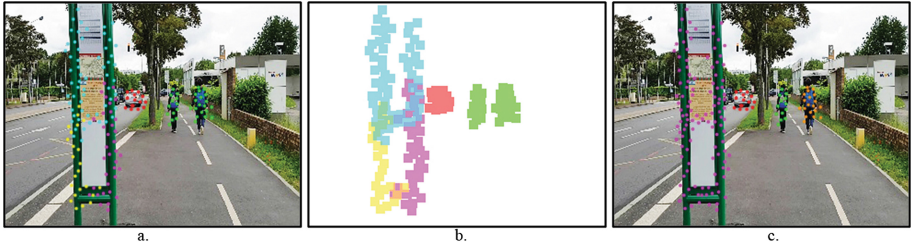


Fig. 3. Motion classes estimation: (a) the interest points clustered using k-means; (b) binary masks associated to each cluster; (c) clusters grouping/division based on spatial principles

points spatial variance within the image. It can be observed that various moving objects existent in the scene can be characterized by the same motion patterns (Fig. 3a). In order to distinguish between dynamic objects with similar motion features (*e.g.* two vehicles approaching or two pedestrians walking in the same direction) we propose to verify the spatial distribution of points within a group. For each cluster we construct a binary mask using the interest points location and its associated region (p_{area}) defined as twice the area of the grid cell used for relevant interest point extraction (Fig. 3b). Clusters satisfying the spatial consistency will contain only interest points that define connected image areas. However, if a cluster forms multiple independent regions it is divided into small classes one for each independent area (Fig. 3c). On the contrary, if two clusters share in common more than 10% of the total region areas the classes are merged together (Fig. 3c). Finally we assign to the background small clusters with less than ten interest points.

Object Degree of Danger Estimation. After objects are identified we need to determine their degree of danger relative to a VI and to classify them. We observed that not all detected obstacles represent a potential risk (*e.g.* far away objects). In this context, we propose defining two areas of proximity, both with a trapezoidal shape, in the near surrounding of a user: one situated on the walking path and the other on the head level (Fig. 4). An obstacle will be considered as having a high degree of danger if it's situated in one of the proximity areas, otherwise it will be marked as non-relevant for the navigation. The size of the proximity areas is depended on the smartphone angle of view (θ) and elevation (E).

If the smartphone has a field of view of 69° , is attached at an elevation of 1.3m and the trapezium height is a third of the image height we can determine the distance from the user and the upper pixel of the trapezium as:

$$D_{object} = 3 \cdot \frac{E}{\tan(\frac{\theta}{2})} = 5.5 \text{ m} \quad (5)$$

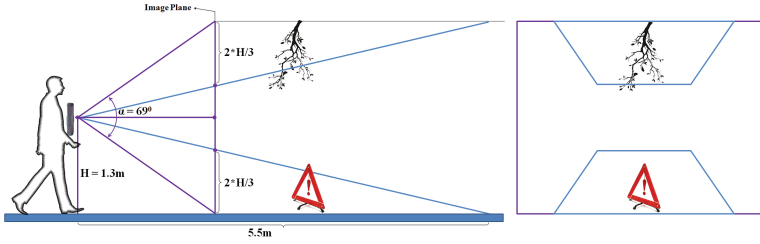


Fig. 4. Obstacles degree of danger estimation based on user proximity areas

The use of the proximity areas prevents our system to continuous launch warning messages for any type of object existent in the outdoor environment. However, the sizes of the trapeziums can be accidentally modified during navigation. In order to prevent accidental errors caused by the device position variation we use the accelerometer sensors existent on the smartphone to determine the device tilt angle. If the tilt angle varies between 60 and 90° the system is considered to function normally. After we established the object location and its degree of danger we need to capture its semantic nature that will helps differentiate between various obstructions situated in the user proximity area. It is important to transmit first an acoustic warning about a vehicle that is approaching VI people rather than an alarm about the presence of a static obstruction such as fence or a garbage can. So, we introduce next a classification framework that will help us better understand the scene and prioritize the set of warnings.

3.2 Obstacle Classification

In order to capture the semantic meaning of all objects existent in the scene, we start the classification framework by constructing offline a training dataset. The training database is divided in four major categories based on their relevance to a VI user as follows: vehicles, bicycles, pedestrians and static objects. In the static obstruction class we included a high variability of instances as: trees, bushes, fences, garbage cans, pylons, traffic signs and lights, mail boxes, stairs. We selected to use only four major categories because our major concern was to develop an efficient classification application that function in real time on a low processing unit. The entire training set is composed of 10000 images selected from the PASCAL repository [27]. We decided to have an equal number of images for each category in order not to advantage a class.

Relevant Interest Point Extraction. Every image patch extracted with the previously described obstacle detection method (Sect. 3.1) is resized so that includes a maximum number of 12 k pixels, while preserving the original aspect ratio. For every patch, we extract interest points using pyramidal FAST [21] algorithm that are further described with BRIEF descriptors [28]. The output of the BRIEF algorithm is a binary vector where each bit is obtained as the

result of intensity test between two adjacent pixels. We could have selected a more powerful interest point descriptor as SIFT [29] or SURF [30], but they prove to be highly expensive in terms of computational time. The main advantage of BRIEF is given by the fact that the descriptor is very fast to compute and compare. Because, the descriptor is a binary vector, the Hamming distance can be here exploited, which is much faster to compute than Euclidian distances between SURF/SIFT descriptors.

Global Image Representation. In order to describe the informational content of an extracted image patch, we propose to develop a global image representation using the VLAD (*Vector of Locally Aggregated Descriptors*) approach [31]. We start with an offline process by constructing a visual codebook $C = \{c_1, c_2, \dots, c_k\} \in \mathbb{R}^{d \times k}$ using the training dataset and the k -means clustering method. We set k to 256 words. Then, for each detected obstacle we extract low level features $D_i = \{d_{i1}, d_{i2}, \dots, d_{in}\} \in \mathbb{R}^{d \times n}$ and we assign every local descriptor to its closest visual word from the vocabulary c_i . The residual r_i for a visual word is computed by accumulating all differences between the local descriptor $d \in D$ and the associated centroid (word in the vocabulary c_i):

$$r_i = \sum_{d \in D; d \cong c_i} d - c_i; \quad i \in 1, \dots, k \quad (6)$$

In order to reduce the influence of frequently occurring components, we applied the power low normalization on the residual vectors: $r'_{i,j} = r_{i,j}^\alpha$. Where $r_{i,j}$ represents the j^{th} element of the residual vector associated to a codebook word c_i . In the experimental evaluation we observed that the optimal value for the α parameter is 0.8. Furthermore, because we want to balance the local descriptors contribution, we apply the residual normalization [32] as follows:

$$r''_i = \frac{r'_i}{\|r'_i\|}; \quad i \in 1, \dots, k \quad (7)$$

The image patch representation using VLAD is obtained as the direct concatenation of all residual vectors (r''_i): $v = [r''_1, r''_2, \dots, r''_k]$. The dimension of a VLAD vector is $p \times k$, where k is the size of the vocabulary and p is the dimension of the binary descriptor. We set the size of BRIEF descriptor to 256. Finally, the vector v is normalized to unit length, which corresponds to:

$$\bar{v} = \frac{v}{\|v\|} = \frac{v}{\sqrt{k}} \quad (8)$$

In order to further reduce the memory requirements we applied the PCA [33] transformation on VLAD vectors. The PCA alleviated the influence of correlated patterns between BRIEF binary components. Hence, we observed that the first 128 components include all the essential information about the feature descriptor, so we performed a dimensionality reduction over VLAD. Finally, the zero-centered vector is binarized to the final vector V by thresholding.

Image Classification. The image classification process can be divided into two parts: an offline process (*i.e.* SVM training) and an online process (*i.e.* SVM prediction). For the SVM training we used the proposed image dataset for which we aimed to determine the optimal decision function that finds a separation hyperplane, between two classes by maximizing the margin:

$$f(V) = \sum_{i=1}^n y_i \cdot \alpha_i \cdot \kappa(V, V_i) + b \quad (9)$$

where $\{(\alpha_i, y_i)\}_{i=1}^n$ is the training set with $y_i \in \{-1, +1\}$, b is the hyperplane free term, α_i is a parameter dependent on the kernel type, while $\kappa(\cdot, \cdot)$ is the selected kernel. In theory $\kappa(\cdot, \cdot)$ can be any reasonable Mercer function, but we observed from our experiments that the chi-square kernel is the most suitable when representing images based on global binary descriptors. The SVM training represents the final step of our offline process. In the online process, for each image patch extracted using the obstacle detection method introduced in Sect. 3.1 we start developing a global representation using the binary VLAD. Then the vector is applied as input to the SVM machine. If $f(V) > 0$ then the example is classified as positive, otherwise it passes to the next SVM machine corresponding to the following category. In order to speed up the decision process the entire online classification method is parallelized on multiple threads, depending on the total number of obstacles present in the analyzed scene.

3.3 Acoustic Feedback

The acoustic feedback improves VI user cognition over the surrounding environment by transmitting warning messages regarding different types of obstacles existent in the scene. In the case of the proposed framework and after discussion with two blind and visually impaired associations, we decided to use the bone conduction headphones technology that satisfies the ears-free constraint and is easy to wear. Moreover, the transmitted sound patterns are carefully designed in order to keep the system intuitive to users without any training phase. The sound patterns are selected not to interfere with other natural environmental sounds. The warning messages are encoded and transmitted in stereo based on the location of a detected obstacle: if the obstruction is situated on the left side of the user the alert message is transmitted on the left helmet and vice-versa for obstruction situated on the right side. For objects positioned on the walking path the VI user will receive the warning message in both helmets.

A final step is represented by the selection of relevant messages to be transmitted depending on the user current situation. In an outdoor navigation scenario, in real urban environments, more potential dangerous objects can be encountered in the near vicinity of a person. To keep the acoustic feedback intuitive and useful we propose to prioritize the warning messages depending on the object type and its distance from the current walking direction of user. The following set of alarms can be generated: “vehicle”, “bicycle”, “pedestrian” and “obstruction”. In order not to confuse the user we decided to launch warning

messages with a frequency rate inferior to two seconds, regardless of the scene dynamics.

4 Experimental Evaluation

4.1 Objective Evaluation

The evaluation of the proposed framework is performed on a database with 20 videos, filmed in real life outdoor environments, with the help of visually impaired people. Each video has an average duration of 10 min, is processed at a resolution of 320×240 pixels and include multiple obstructions either static or dynamic. In addition, because the acquisition process is performed using the video camera embedded on a regular smartphone, attached to the VI user with the help of a chest mounted harness all videos are characterized by various type of background/camera motion, include abrupt changes in the light intensity, are trembled or cluttered. Each frame of the video stream was annotated by a set of human observers. Once the ground truth dataset is available the performances of the detection and the classification modules can be globally estimated using two error parameter, denoted $M_{D/C}$ and $F_{D/C}$ representing the number of missed detected/classified obstacles and the number of false alarms (false detected/classified obstructions). Finally, $N_{D/C}$ represents the total number of dynamic/static objects correctly detected/classified by our module. In order to determine globally the performances of each module we used for evaluation the traditional metrics as precision (P), recall (R) and $F - score$ defined as:

$$R = \frac{N_{D/C}}{N_{D/C} + M_{D/C}}; P = \frac{N_{D/C}}{N_{D/C} + F_{D/C}}; F - score = \frac{2 \cdot P \cdot R}{R + P} \quad (10)$$

Table 1 presents the experimental results obtained for the obstacle identification module. As it can be observed, the average $F - score$ for all types of obstacles is 85%. Particularly better results are obtained for vehicles due to the distinctiveness of the motion vector magnitudes. In Table 2 we present the performance of the classification module in terms of precision, recall and $F - score$ when applying as input the image patches extracted by detection system. The experimental results validate our approach, with recognition scores superior to 82% for every category. In particular, the lowest results are obtained for the Bicycles category because when we trained the SVM most of the bikes were static without any human hiding them. In our case the detection module, returns as input to the classification system a patch containing both the bikes and the human.

In Fig. 5 we give a graphical representation of the experimental results obtained by our obstacle detection and classification modules. In all cases, the video camera is characterized by an important variation caused by the subject displacement. We marked the detected objects with rectangles of different colors in order to differentiate between various elements existent in the scene. Our system is able to detect static hazards such as: road signs, pylons or bushes

Table 1. Experimental results of the obstacle localization module

	Ground truth	N_D	M_D	F_D	Recall	Precision	$F - score$
Vehicle	568	505	63	42	0.89	0.92	0.9
Pedestrians	448	381	67	58	0.85	0.86	0.85
Bicycles	210	175	35	24	0.83	0.87	0.85
Obstructions	402	344	58	41	0.85	0.89	0.87

Table 2. Experimental results of the obstacle classification module

	Ground truth	N_C	M_C	F_C	Recall	Precision	$F - score$
Vehicle	505	464	41	24	0.91	0.95	0.93
Pedestrians	381	351	30	27	0.92	0.92	0.92
Bicycles	175	144	31	8	0.82	0.94	0.88
Obstructions	344	308	36	22	0.89	0.93	0.91

based situated on head or foot level using only the camera apparent motion. However, the method will not identify all static obstacles existent in the scene, but just the one situated on the VI user walking path. This behavior does not penalize the performance of the application since users are not interest in any type of obstruction present in an outdoor scenario, but only about objects that could affect the safety navigation. Regarding the dynamic obstacles, they can be identified at larger distances from the subject (superior to ten meters) due to the more important magnitude of the associated motion vectors. The method is able to correctly classify pedestrians, bikes or vehicles with high $F - score$ rates. However, in the case of pedestrians, which are non-rigid objects, characterized by various types of motions, in some situations the object is divided into multiple unconnected parts (Fig. 5). In this case, the classification method will receive as input only parts of the subject and may return incorrect results. In terms of the computational time, when implementing the entire framework on a regular smartphone device (Samsung S7) running Android as an operating system we obtain an average processing speed of 100 ms per image for both the obstacle detection and classification modules. In this context the processing speed is around 10 fps.

4.2 Subjective Evaluation

In the following part of the experimental evaluation we were focused on determining the VI users' degree of satisfaction after using the proposed system prototype. The main objective of the testing phase was to determine if the users can: start the application, walk safe in an outdoor environment, avoid collision and acquire sufficient additional knowledge over the environment. The participant was asked to complete a pre-established route in two scenarios: the first

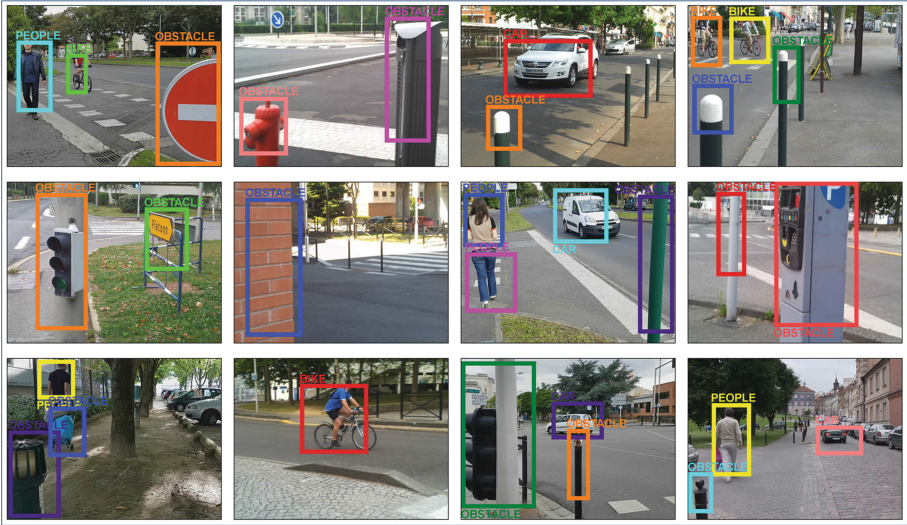


Fig. 5. Experimental results of the combined framework: obstacle detection and classification (Color figure online)

one assumes navigating using only the white cane and the second when combining the white cane with our system as an assistive device. After the task was finished, an observer conducted an interview and each participant was asked a set of questions about their impressions over the device.

The following conclusions can be highlighted: (1) some users, because of their resilience and mistrust of new technologies felt insecure to use electronic assistive devices. In their opinion it is important to develop a system designed not to replace the cane, but to complement it with additional functionalities. (2) The users considered the system very useful and easy to worn and understand, but an initial training phase is required in order to understand all functionalities. However, VI users already manipulating smartphone devices expressed strong interest on this type of application. (3) The acoustic feedback is transmitted fast enough in order to avoid dangerous situations. The VI considered that bone-conduction headphone is appropriate to wear because it does not impede the ambient sound cues.

5 Conclusions and Perspectives

In this paper we introduced a blind and visually impaired navigational assistant, designed to detect and recognize both static and dynamic obstacles encountered by visually impaired users during outdoor navigation. In contrast to prior state of the art methods, our technique does not require any information about the obstacle type and position and was designed to achieve real-time capabilities on

a smartphone device. The output of the recognition module is transformed into set of warning messages transmitted to the VI users through acoustic feedback.

The evaluation of our framework was performed on a video corpus, with 20 elements, acquired with the help of VI users and depicting urban outdoor environments. The system proves to be robust to important camera/background motion or to changes in the illumination. The video stream is processed with an average speed of 10 fps (on a Samsung S7 device), while the warning messages are transmitted fast enough so that user walks normally. In addition, we introduced a subjective evaluation over the system by presenting the VI people degree of satisfaction and comments after using our prototype.

For further work and implementation we proposed integrating in our framework additional functionalities, such as: guided navigation, face recognition capabilities and a shopping assistance in supermarkets. Moreover, with the development of the smartphone industry, the 3D video cameras will be available shortly on commercial devices (*e.g.* Lenovo Phab 2 Pro). In this context we envisage better estimating the distances between obstacles and VI people by using depth information.

Acknowledgement. This work was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS - UEFISCDI, project number: PN-II-RU-TE-2014-4-0202.

References

1. Blasch, B.B., Wiener, W.R., Welsh, R.L.: Foundations of Orientation and Mobility, 2nd edn. American Foundation for the Blind, New York (1997)
2. Golledge, R.G., Marston, J.R., Costanzo, C.M.: Attitudes of visually impaired persons towards the use of public transportation. *J. Vis. Impairment Blindness* **90**, 446–459 (1997)
3. Johnson, L.A., Higgins, C.M.: A navigation aid for the blind using tactile-visual sensory substitution. In: 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 6289–6292 (2006)
4. Sainarayanan, G., Nagarajan, R., Yaacob, S.: Fuzzy image processing scheme for autonomous navigation of human blind. *Appl. Soft Comput.* **7**(1), 257–264 (2007)
5. Yu, J., Chung, H.I., Hahn, H.: Walking assistance system for sight impaired people based on a multimodal information transformation technique. In: ICCAS-SICE, pp. 1639–1643 (2009)
6. José, J., Farrajota, M., Rodrigues, J., Buf, J.D.: The smart vision local navigation aid for blind and visually impaired persons. *Int. J. Digital Content Technol. Appl.* **5**, 362–375 (2011)
7. Lin, Q., Hahn, H., Han, Y.: Top-view based guidance for blind people using directional ellipse model. *Int. J. Adv. Robot. Syst.* **1**, 1–10 (2013)
8. Peng, E., Peursum, P., Li, L., Venkatesh, S.: A smartphone-based obstacle sensor for the visually impaired. In: Yu, Z., Liscano, R., Chen, G., Zhang, D., Zhou, X. (eds.) UIC 2010. LNCS, vol. 6406, pp. 590–604. Springer, Heidelberg (2010)
9. Manduchi, R.: Mobile vision as assistive technology for the blind: an experimental study. In: Miesenberger, K., Karshmer, A., Penaz, P., Zagler, W. (eds.) ICCHP 2012, Part II. LNCS, vol. 7383, pp. 9–16. Springer, Heidelberg (2012)

10. Tapu, R., Mocanu, B., Bursuc, A., Zaharia, T.: A smartphone-based obstacle detection and classification system for assisting visually impaired people. In: IEEE International Conference on Computer Vision Workshops (ICCVW), pp. 444–451 (2013)
11. Dakopoulos, D., Bourbakis, N.: Preserving visual information in low resolution images during navigation of visually impaired. In: Proceedings of the 1st International Conference on Pervasive Technologies Related to Assistive Environments, pp. 1–27 (2008)
12. Saez, J.M., Escolano, F., Penalver, A.: First steps towards stereo-based 6DoF SLAM for the visually impaired. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), p. 23 (2005)
13. Pradeep, V., Medioni, G., Weiland, J.: Robot vision for the visually impaired. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 15–22 (2010)
14. Saez, J.M., Escolano, F.: Stereo-based aerial obstacle detection for the visually impaired. In: Workshop on Computer Vision Applications for the Visually Impaired (2008)
15. Schauerte, B., Koester, D., Martinez, M., Stiefelwagen, R.: Way to go! detecting open areas ahead of a walking person. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014 Workshops. LNCS, vol. 8927, pp. 349–360. Springer, Heidelberg (2015)
16. Khan, A., Moideen, F., Lopez, J., Khoo, W.L., Zhu, Z.: KinDectect: kinect detecting objects. In: Miesenberger, K., Karshmer, A., Penaz, P., Zagler, W. (eds.) ICCHP 2012. LNCS, vol. 7383, pp. 588–595. Springer, Heidelberg (2012). doi:[10.1007/978-3-642-31534-3_86](https://doi.org/10.1007/978-3-642-31534-3_86)
17. Takizawa, H., Yamaguchi, S., Aoyagi, M., Ezaki, N., Mizuno, S.: Kinect cane: an assistive system for the visually impaired based on three-dimensional object recognition. In: IEEE/SICE International Symposium on System Integration (SII), pp. 740–745 (2012)
18. Brock, M., Kristensson, P.: Supporting blind navigation using depth sensing and sonification. In: Proceedings of the ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication, pp. 255–258 (2013)
19. Panteleris, P., Argyros, A.A.: Vision-based SLAM and moving objects tracking for the perceptual support of a smart walker platform. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8927, pp. 407–423. Springer, Heidelberg (2015). doi:[10.1007/978-3-319-16199-0_29](https://doi.org/10.1007/978-3-319-16199-0_29)
20. Li, W., Li, X., Goldberg, M., Zhu, Z.: Face recognition by 3D registration for the visually impaired using a RGB-D sensor. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8927, pp. 763–777. Springer, Heidelberg (2015). doi:[10.1007/978-3-319-16199-0_53](https://doi.org/10.1007/978-3-319-16199-0_53)
21. Tuzel, O., Porikli, F., Meer, P.: Region covariance: a fast descriptor for detection and classification. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 589–600. Springer, Heidelberg (2006)
22. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proceedings of Fourth Alvey Vision Conference, pp. 147–151 (1988)
23. Mikolajczyk, K., Schmid, C.: Scale and affine invariant interest point detectors. *Int. J. Comput. Vis.* **60**, 63–86 (2004). Ubiquitous Intelligence and Computing SE - 45
24. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proceedings of the 7th International Joint Conference on Artificial Intelligence, vol. 2, pp. 674–679 (1981)

25. Lee, J., Kim, G.: Robust estimation of camera homography using fuzzy RANSAC. In: Gervasi, O., Gavrilova, M.L. (eds.) ICCSA 2007. LNCS, vol. 4705, pp. 992–1002. Springer, Heidelberg (2007). doi:[10.1007/978-3-540-74472-6_81](https://doi.org/10.1007/978-3-540-74472-6_81)
26. Hamerly, G., Elkan, C.: Learning the k in k-means. In: Neural Information Processing Systems (2003)
27. Everingham, M., Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **88**, 303–338 (2010)
28. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: BRIEF: binary robust independent elementary features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 778–792. Springer, Heidelberg (2010)
29. Lowe, D.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **60**, 91–110 (2004)
30. Bay, H., Ess, A., Tuytelaars, T., Gool, L.V.: Speeded-up robust features (SURF). *Comput. Vis. Image Underst.* **110**(3), 346–359 (2008)
31. Jegou, H., Douze, M., Schmid, C.: Product quantization for nearest neighbor search. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(1), 117–128 (2011)
32. Delhumeau, J., Gosselin, P.H., Jegou, H., Perez, P.: Revisiting the VLAD image representation. In: Proceedings of the 21st ACM International Conference on Multimedia, pp. 653–656 (2013)
33. Zou, H., Hastie, T., Tibshirani, R.: Sparse principal component analysis. *J. Comput. Graph. Stat.* **15**(2), 265–286 (2006)