

An Architecture for Collision Risk Prediction for Visually Impaired People

Natal Henrique Cordeiro*[†]

*Federal Institute of São Paulo

Votuporanga, SP, Brazil

Department of Computing

Email: natalhenrique@ifsp.edu.br

Emerson Carlos Pedrino[†]

[†]Federal University of São Carlos

São Carlos, SP, Brazil

Department of Computer Sciences

Email: emerson@dc.ufscar.br

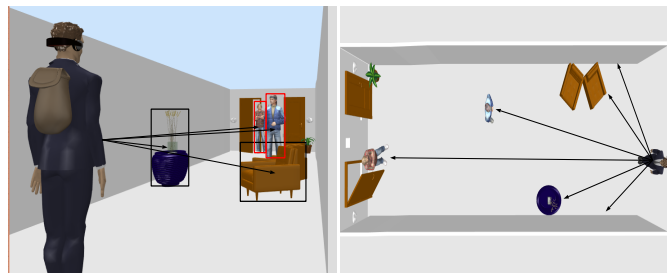
Abstract—The production of sensory substitution equipment for the visually impaired (VIP) is growing. The aim of this project is to understand the VIP context and predict the risks of collision for the VIP, following an analysis of the position, distance, size and motion of the objects present in their environment. This understanding is refined by data fusion steps applied to the Situation Awareness model to predict possible impacts in the near future. With this goal, a new architecture was designed, composed of systems that detect free passages, static objects, dynamic objects and the paths of these dynamic objects. The detected data was mapped into a 3D plane verifying positions and sizes. For the fusion, a method was developed that compared four more general classifiers in order to verify which presented greater reliability in the given context. These classifiers allowed inferences to be made when analyzing the risks of collision in different directions. The architecture designed for risk prediction is the main contribution of this project.

I. INTRODUCTION

According to the World Health Organization (WHO) report [1], there were an estimated 285 million people with visual impairment in the world in 2014. Visually impaired people (VIP) have mobility problems and this has a major impact on social inclusion. The research and consequent development of sensory support and navigation technologies for visually impaired people (VIP) is increasing [2]–[8]. These systems are designed to provide locomotion based on features extracted from the environment.

VIPs require technologies that go beyond just indicating the desired destination; these technologies must recognize patterns, contextualize aspects of the environment and indicate what action should be taken to ensure the safety of the VIP. Due to the difficulties that the VIPs have with orientation in specific environments, this project presents a system composed of computer vision techniques and analysis of images that provide data for the detection of obstacles present in the scene also informing their locations, their distances, their movements and directions. Fig. 1 presents a simulation of an indoor environment containing the VIP and various obstacles.

This Sensory Analysis System For Visually Impaired People (SAS-VIP) introduces a new data fusion architecture based on contexts similar to those established by a person with vision to support VIP decision making. According to Zhu *et al.* [9], a context refers to current values and specific data that provide the user with an activity or situation.



(a) Simulation (Indoor environment) (b) Simulation with different angle

Fig. 1. (a) Indoor environment containing the VIP and various obstacles. (b) The same environment (a) from a different angle

Contributions: In an environment there may be a set of contexts. A context can be formed through the relationship that entities have with the goal that a user must achieve. An entity can be an object, a person, or an area Zhu *et al.* [9]. If any entity is considered relevant to VIP traffic without collisions, it can be chosen to form a context. In this way, this paper presents a new architecture that predicts collision risks for usual contexts (with obstacles) present in indoor environments and this is the main contribution of this project. This architecture was designed based on the Salerno Model for high-level fusion. According to Liggins *et al.* [10], this model incorporates concepts from the Joint Directors of Laboratories (JDL) model and the SAW model proposed by Endsley [11].

Locomotion in unknown environments that contain static objects (SO) and dynamic objects (DO) is one of the main difficulties that VIP must deal with. Several technological solutions have been presented in order to assist the VIP to find out their position, which elements are in their way and where is the safest place to move around. In most of these architectures, different types of sensors were used, such as those that detect distance, presence, motion and colors [2], [5], [6], [12]–[14]. Typically these sensors are used as data sources. However, the systems do not have progressions to more refined fusion phases in order to correct errors, remove redundancies and generate decisions that a human being can trust.

Other applications have been produced [15]–[18] to provide alternative directions for the VIP but few have attempted a

deeper analysis of the objects present in the scene, which can perform data fusion and make decisions based on the Situation Awareness (SAW) level of projection. This project fits into contexts where data fusion is applied as a fundamental technique in decision making and aims to reach higher data-fusion levels in order to forecast possible collisions in the near future. Thus, the main contribution of this project is the development of an architecture that performs the modeling of a dataset used to make inferences. These inferences are then applied in order to analyze the risks of collision in all directions that the VIP might move.

This paper is divided into six sections. Section I has already presented a brief contextualization of the VIP problem and the contributions that this project can make. In Section II, a comparison is made between the most recent related works found in the literature and the present project to see what possible contributions this study will provide for VIP mobility and for the state of the art architecture. Section III presents the proposed architecture that predicts collision risks in indoor environments. Section IV presents the experiments performed using the concepts covered in Section III. In Section V, the results obtained from comparing different classifiers are discussed. Section VI presents the final considerations.

II. RELATED WORK

There are projects that apply different techniques to solve specific navigation problems. Some projects ([16], [7], [19], [20]) detect nearby obstacles and indicate a region with a potential collision. Other projects use equipments such as tags or code labels (RFID or QR-Code) ([4], [3], [21]) that are installed in the environment. This strategy is interesting to provide the VIP with the appropriate place to move or recognize objects but it is not suitable to inform possible collisions with dynamic objects or obstacles that do not have RFID. This facility requires prior planning at all locations where the VIP will be traveling. SAS-VIP was designed not to rely on technologies installed in the environment or remotely requested information. Another important feature that should be emphasized is the dependency of installing multiple RFIDs in the environment. If there is no RFID in a certain place, the system does not have basic information for decision making and the system loses its purpose. Thus, systems using technologies such as RFID, QR Codes or any other type of remote information, should be integrated with other systems that complement the mobility needs of VIP so that they don't depend only on sources of data implanted a priori.

In the study by Pundlik *et al.* [22], an RGB camera and gyro-sensor are used to analyze the optical flow, check camera movements and detect possible nearby static and dynamic objects. Their project has a similar objective to that of the present work, referring to the detection of obstacles, but it uses different sensors and techniques. When only an RGB camera is used, the exact position of an obstacle cannot be calculated, which makes it difficult to indicate a reliable collision-free direction. The RGB camera also has many problems related to lighting. However it is an alternative for outdoor environments.

The infra-red camera used in this project allows reliable calculations of distances on the X, Y and Z axes and has no lighting restriction to its indoor use. However, the techniques implemented in this project are specific to closed environments due to the exhaustive use of the IR sensor. Pooling these two studies could make a significant contribution to VIP equipment for use in different environments.

Joseph *et al.* [23] developed a system that adopts the Kinect motion sensor, to obtain a disparity map and addresses the use of social sensors to provide data from sites and social networks that support VIP to find out if there is any danger to mobility in a given region. With these data sources, the SAW can be generated, which allows the production of more reliable information. However, Joseph's system depends on data feeds provided by others. In the context of support systems for VIPs, there is nothing in the literature that describes an architecture containing computer vision techniques and embedded machine learning to establish data fusion at a level that predicts collisions or accidents between a VIP and environmental elements. An important feature of this project is to provide the VIP with the possibility of making inferences to analyze the collision risk.

III. SENSORY ANALYSIS SYSTEM FOR VISUALLY IMPAIRED PEOPLE (SAS-VIP)

For the development of the architecture proposed here, first we analyzed the different ways of grouping the set of characteristics of the detected stationary objects (SO) and dynamic objects (DO). The process of detecting these objects can be seen in Fig. 2, where the moments of acquisition of these data are presented through images.

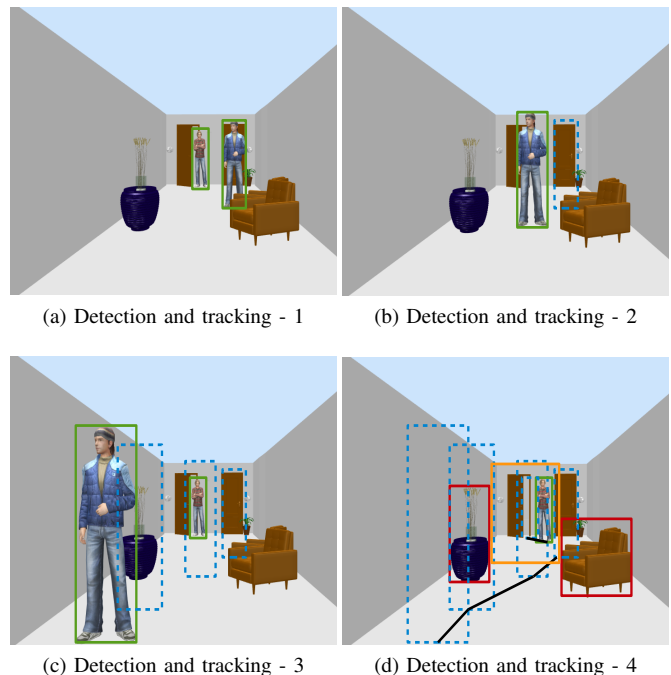


Fig. 2. Indoor environment with obstacles

The green contours in Fig. 2 represent the dynamic objects. The blue dashed outlines in the images 2b, 2c, 2d, of the same figure, illustrate some positions that are stored and which allow for the generation of the path followed by DO. The red and orange outlines of image 2d display the SO and free passage (FP) respectively. In the same figure, the black line shows the route taken by a DO.

For authentic analysis of all the obstacles and information present in the environment, the objects had to be mapped in a three-dimensional plane with real world coordinates. For this to be done, all distances were converted to millimeters.

Based on these concepts, the SAS-VIP architecture was then developed using SAW. The SAS-VIP architecture (Fig. 3) consists of three modules: an Input and Output Module (I/O) (section III-A); a Vision Module (VM) (section III-B); and a Fusion Module (FM) (section III-C).

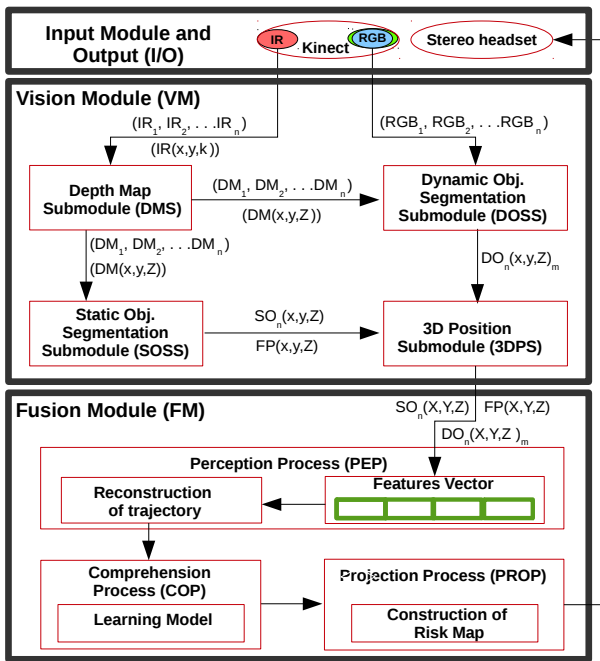


Fig. 3. SAS-VIP architecture

A. Input Module and Output (I/O)

The SAS-VIP Input and Output Module (Fig. 3) consists of a Kinect sensor as input and a stereo headset as output. The Kinect basically has three sensors; an accelerometer and two video cameras (Infrared (IR) and RGB).

In this work, we only use the IR and RGB camera. The objective of the IR is to provide frames in real time, containing scene depth maps, for the Vision Module, which will be used to detect static and dynamic objects present in the environment.

The RGB camera was used to differentiate between the movement of the dynamic object and the camera movement. Finally, in this module, there is feedback to the VIP through sounds (s) or beeps (b) provided by the Fusion Module for orientation.

B. Vision Module (VM)

The VM is composed of four submodules, which together are responsible for the extraction of characteristics, such as the 3D position of static and dynamic objects and the direction of free passages.

1) *Depth Map Submodule (DMS)*: The architecture presented in this article uses a Microsoft Kinect RGB-D as the acquisition sensor (AS). In this submodule, given the frames (30 fps) ($IR_i, i=1..30$) from the AS, its color maps for the 3D distances of the scene objects are converted into appropriate matrices ($DM_i, i=1..30$) with values in millimeters (mm) associated with these values (x,y,k), where (x,y) represents the image coordinates and k the intensity value (RGB channels) from the VIP. A method was then developed that converts the value k to millimeters, generating the value Z (in millimeters).

2) *Static Object Segmentation Sub-module (SOSS)*: This submodule separates the static objects present in the scene that are less than 2250_{mm} (millimeters) away from the VIP, in order to avoid collisions. Also, the direction of free passage is provided. Segmentation of these static objects is accomplished by means of a Threshold ($SOSS_{so}$) that scans the depth image ($DM(x,y,Z)$) eliminating the background where Z is more than 2250_{mm} . This filter ($SOSS_{so}$) means that only the obstacles in the foreground and which generate greater risks of impact in a short period of time are analyzed.

The direction of free passage is also detected in SOSS. To find out the free passage, another Threshold ($SOSS_{fp}$) filter was applied to the depth image ($DM(x,y,Z)$) excluding data where Z is greater than 3000_{mm} .

Distances have been predefined based on the information needed by a VIP to move safely. A VIP walking, on average, at 0.75 m/s, needs to know which obstacles pose the greatest risk of collision. At this speed, the VIP has approximately 2.5 seconds to bypass close obstacles and approximately 4 seconds to proceed in the direction of a detected free passage until the system detects a new FP. These measures were used in the experiments after conducting interviews with VIP following the guidelines of the GDTA (SAW model) [24].

An algorithm (Algorithm 1) was developed to present the data flow of the architecture (see Fig. 3) with respect to detection of SO and FP. Each instruction has a previous explanation.

The SOSS, was implemented using the thresholding techniques based on the depth map. In it, only the IR sensor is used as the AS (line 1) in Algorithm 1. The DMS receives data supplied by the IR sensor (line 2) and generates the depth map containing the distances of all the coordinates ($DM(x,y,Z)$ shown in lines 3 and 4). The SOSS, in turn, applies thresholding techniques to detect the static objects (line 5) and free passages (line 6). Thus, in lines 7 and 8), their contours and their centers of mass respectively (lines 9 and 10) are calculated. To generate the edge, a threshold filter was used, followed by the Canny edge detection filter to eliminate remaining noise and, consequently, produce a better quality result for the area calculation. When the detector provides a

region bounded by a curve or a closed contour, its area can be calculated.

The 3D position of an object is calculated based on its center of mass but, for each contour performed on the detected object, the (x, y) coordinates of the beginning and of the end of the contour are stored. In this way, the widths and heights are made available.

In lines 11 and 12) 3DPM refer to the centers of mass ($SO_n(x, y, Z)$, $FP(x, y, Z)$) with the distance (Z) already calculated in millimeters (line 11 and 12). The 3DPM is then responsible for converting the values (x, y) in pixel to millimeters and providing the 3D position of the SO and the FP (line 13) to be mapped onto a single reference system. Finally, the 3DPM submits the beeps at the calculated position in millimeters to the stereo headset.

Initially a method was used that emits a beep produced specifically by means of the distance to the detected object. The beeps for static objects are produced from the left side of the headset and the dynamic objects on the right side.

Algorithm 1: DETECTION OF STATIC OBJECTS

Input: $Video_{IR}$
Output: SO and FP detected

```

1 while (Capture  $Video_{IR}$ ) do
2    $((IR_1, IR_2, \dots, IR_n)) \leftarrow \text{convertFrame}(Video_{IR})$ 
3    $\Theta \leftarrow DM_{frames}((IR_1, IR_2, \dots, IR_n(x, y, k)))$ 
4    $((DM_1, DM_2, \dots, DM_n(x, y, Z))) \leftarrow \Theta$ 
5    $SO_{Segment} \leftarrow \text{Segmentation}(\text{threshold} < 2250_{mm}, IR_1)$ 
6    $FP_{Segment} \leftarrow \text{Segmentation}(\text{threshold} > 3000_{mm}, IR_1)$ 
7    $\text{findContours}(SO_{Segment}, SO_{Contour})$ 
8    $\text{findContours}(FP_{Segment}, FP_{Contour})$ 
9    $SO(x, x, Z) \leftarrow \text{CMW}(SO_{Contour})$ 
10   $FP(x, x, Z) \leftarrow \text{CMW}(FP_{Contour})$ 
11   $SO(X, Y, Z)_{mm} \leftarrow 3DPM(SO(x, x, Z))$ 
12   $FP(X, Y, Z)_{mm} \leftarrow 3DPM(FP(x, x, Z))$ 
13 return  $SO(X, Y, Z)_{mm}, FP(X, Y, Z)_{mm}$ 

```

3) *3D Position Submodule (3DPS)*: The 3D Position Submodule (3DPS) is responsible for converting the reference system formed in pixel(x, y) and millimeters (Z) to a reference system (X, Y, Z) based only in millimeters. This mapping is performed in three dimensions using the same reference system (millimeters) from the DO position ($DO_n(x, y, Z)_m$).

The conversion of a pixel present in the 2D plane to the 3D is performed by means of measurements that correspond to the pixel size(x, y) in the distance (Z) in which it lies. It is emphasized that the distance was converted to millimeters (Z) by DM. All relevant objects had their positions converted from pixels to millimeters using Equations 1 and 2.

$$Dx_{mm} = x_{px} * \frac{Dz_{mm}}{Sx_{px}} \quad (1)$$

$$Dy_{mm} = y_{px} * \frac{Dz_{mm}}{Sy_{px}} \quad (2)$$

- The unknown variables x_{px} and y_{px} represent the position of the object in the image coordinates (in pixels) on the x-axis and the y-axis respectively.
- Dz_{mm} = Distance of the object in millimeters.
- Sx_{px} and Sy_{px} = The amount of pixels in the image on the respective axes (640 x 480).
- In this way, the position of the object in millimeters on the x-axis (Dx_{mm}) and on the y-axis (Dy_{mm}) is calculated.

4) *Dynamic Object Segmentation Submodule (DOSS)*:

The DOSS extracts features related only to moving objects present in the VIP field of view. Acquiring and relating these characteristics to the environment is important for the data fusion system in providing a safe direction for VIP movement.

Another algorithm (Algorithm 2) was developed to present the data flow of the architecture (see Fig. 3) with respect to detection of DO. Each instruction has a previous explanation. In this algorithm, the IR and RGB sensors are both used as the AS (line 1). The DMS receives data supplied by the IR sensor (line 2) and generates the depth map containing the distances of all the coordinates ($DM(x, y, Z)$ shown in lines 3 and 4). The DOSS receives data supplied by the RGB and IR sensor (lines 5 and 7).

For the dynamic object detection process, Farneback Optical Flow ($Farneback_{OF}$) was used to differentiate between the movement of the dynamic object and the camera movement (see line 6) but, after detection, the tracking is performed by background subtraction (line 7). In this way, when the system detects a dynamic object, the system issues an alert for the VIP to slow down so that the BS will better detect the path of the dynamic object.

The DOSS was developed using the Mixture of Gaussians technique to connect components. This technique was developed by reducing the sensitivity in the BS process, after increasing the Threshold ($varThreshold$). This threshold is the quadratic distance between the pixel and the sample (segmented object) used to decide if the pixel is near it. In the segmentation process for dynamic objects, being more complex, a minimum area for contour generation was defined, as well as the application of smoothing, dilation and erosion filters.

After detecting the dynamic object, its contour (line 8) and its center of mass (line 9) were calculated. CMW refer to the centers of mass ($DO(x, y, Z)$) with the distance (Z) already calculated in millimeters. The 3DPM is then responsible for converting the values (x, y) in pixel to millimeters and providing the 3D position of the DO (line 10) to be mapped onto a single reference system. Finally, the 3DPM submits (line 11) the beeps at the calculated position in millimeters to the stereo headset.

The DO analysis can produce some different information about safer paths to follow. Although a moving object produces a high collision risk but, at the same time, its trajectory may indicate a safer region for the VIP to move in in the future.

Algorithm 2: DETECTION AND TRACKING OF DYNAMIC OBJECTS USING $Farneback_{OF}$ AND BS

Input: $Video_{IR}$ and $Video_{RGB}$

Output: Detection and tracking of DO

```

1 while (Capture  $Video_{IR}$  and  $Video_{RGB}$ ) do
2    $((IR_1, IR_2, \dots, IR_n)) \leftarrow \text{convertFrame}(Video_{IR})$ 
3    $\Theta \leftarrow DM_{frames}((IR_1, IR_2, \dots, IR_n(x, y, k)))$ 
4    $((DM_1, DM_2, \dots, DM_n(x, y, Z))) \leftarrow \Theta$ 
5    $((RGB_1, RGB_2, \dots, RGB_n)) \leftarrow \text{convertFrame}(Video_{RGB})$ 
6   if  $Detection_{DO}([Farneback_{OF}], RGB_1)$  then
7      $DO_{segment} \leftarrow \text{Segmentation}([BS], varThreshold = 150, DM_1)$ 
8      $findContours(DO_{segment}, DO_{contour})$ 
9      $DO(x, x, Z) \leftarrow CMW(DO_{contour})$ 
10     $DO(X, Y, Z)_{mm} \leftarrow 3DPM(DO(x, x, Z))$ 
11 return  $DO(X, Y, Z)_{mm}$ 

```

C. Fusion Module (FM)

The FM aims to generate contextualized information to provide safe, reliable mobility for the VIP. The FM is composed of three processes: the Perception Process (PEP), the Comprehension Process (COP) and the Projection Process (PROP). Each process belonging to the FM is intended to provide the information needed to make the decisions.

1) *Perception Process (PEP)*: This process analyzes the characteristics of relevant objects in the scene. The contextualization of these objects provides the requirements for the highest levels of abstraction of the SAW. Fig. 4 represents the operation of the architecture, which aims to analyze the collision risks in different directions (DR) related to different distances (DS). The architecture was based on the features extracted from Fig. 2 and then a representative map (X, Z) was produced containing the directions on the X axis and distances on the Z axis.

Through intervals and intersections between DR and DS, fields were generated (Map (DR, DS)) that indicate the position of important decision-making instants. These instances are represented by circles with the following colors: 1- Static objects (red); 2- Dynamic objects (green); 3- Paths created by DO (Blue); 4- Free passage (Yellow).

The map in Fig. 4, shows that in directions 1,2 and 3 (red circles) there is an SO at a distance of 2000_{mm} . This obstacle is wide and, because of this, it appears in directions 1 to 3 as well. Next to it, in directions 4 to 10 (yellow circles), there is an FP 3000_{mm} away. The directions 6 to 8 (blue circles, Fig. 4), represent a path already performed by another dynamic object at a distance of 4500_{mm} , ending its movement in directions 3 to 5, but only 1250_{mm} away from the DV. The green circles, a dynamic object is detected in real time but at more than 4000_{mm} distance.

By calculating the position and size of the obstacles, they could be mapped in 3D and the dataset that makes up an inference analyzed empirically. It should be noted that, to calculate the position and size of the objects, equations 1 and

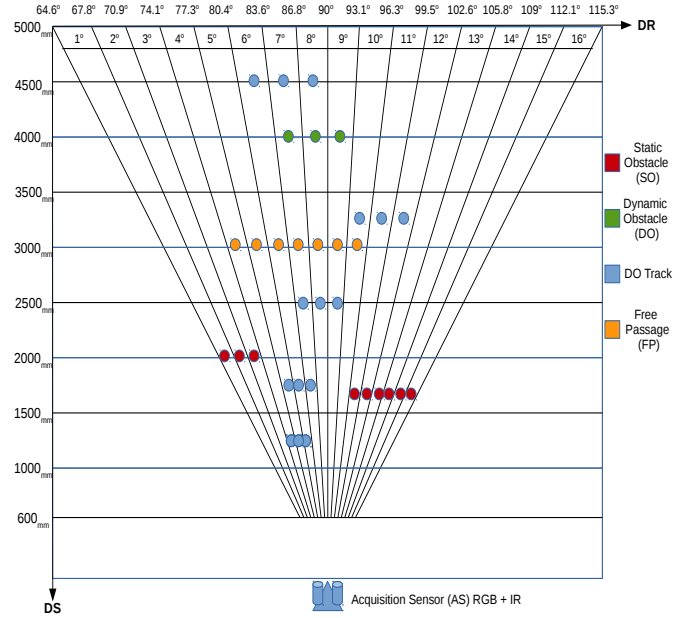


Fig. 4. 3D Map

2 presented in subsection III-B3 were used.

2) *Comprehension Process (COP)*: The Comprehension Process (COP) performs the second phase of the FM. In the COP, a new data fusion is performed, but using information that has already been refined by the above process. Analysis of objects is carried out, based on their behavior, actions, intentions, relevance and capability [10].

The architecture in this project shows that all detected elements can be mapped in real time. After collecting a set of mapped data, a database was created to classify the collision risk probabilities present in each direction. Thus, it was possible to generate a learning model that would allow inferences to be made.

In order to check the possibilities of collision risks in each direction, it was necessary to contextualize: the position of the SO and the DO, the position of the FP and the paths followed by the DO. With this data mapped in a 3D plane, it was possible to classify for each direction the following collision risks: 1- Low; 2- Moderate; 3- High; 4-VeryHigh. Table I shows clearly the composition of the database that generated the learning model.

For every direction where an obstacle was detected, the distance to the FP or path made by DO was stored in millimeters, which allowed the definition of the collision risk. The symbol ? was inserted in situations where no information was detected. The dataset created has more than one hundred (100) lines as shown in Table I (Lines 1-8). This table is just an example of a part of the dataset

The created dataset has more than one hundred (100) different sets of data as shown in Table I (Lines 1 to 8). The model tries to balance all attributes of the relationship.

3) *Projection Process (PROP)*: The Projection Process (PROP) is the third phase of the FM. This process needs

TABLE I
TRAINING DATASET

Example	SO_{mm}	DO_{mm}	FP_{mm}	$Paths_{mm}$	class
1	?	4530	3005	2030	low
2	2200	2180	?	720	moderate
3	850	1610	?	700	high
4	1640	980	?	1600	veryhigh
5	?	?	3150	600	low
6	1700	2425	?	630	moderate
7	?	1350	?	980	high
8	?	760	?	1065	veryhigh

to merge the information produced by the COP in order to sketch situations in the near future. The PROP projects the actions of objects in relation to the VIP to define the possible collisions for a given route. Inferences with elements and current situations are made that project potential collisions.

As already seen for COP, the learning process was carried out to create a model that would allow inferences to be made. In this process (PROP), the inferences are executed in all the directions specified in Fig. 4. The possibility of moving in a safer direction is also verified, after the inferred results, since a set of 3 consecutive directions is required with little collision risk. This is because the width of the space a person will occupy is more than one direction.

IV. EXPERIMENTS

The VM module of the SAS-VIP architecture provided the basic features for the four navigation systems. The VM was written in the C++ language with the help of computational vision and real-time image processing techniques from the OpenCv [25] library. For the experiments, the Kinect device was placed at the front of the waist of the VIP. Inside a backpack, there was an auxiliary battery to power the sensor and a small computer for processing the system.

All the characteristics were obtained indoors with static and dynamic objects of different sizes, positions and lighting (see Fig. 5 and 6). For the moving objects, the direction and the average speed during the period of detection were also modified. Experiments were carried out with all the processes of SAS-VIP architecture to validate the information that the Fusion module needs to provide inferences for a given context.

A. Static Object Segmentation Submodule

This submodule separates the static objects present in the scene that are less than 2250_{mm} away from the VIP, in order to avoid collisions. Also, the direction of free passage is provided. Segmentation of these static objects (see item 5c of Fig. 5) is accomplished by means of a Threshold (line 5, Alg. 1) that scans the depth image ($DM(x,y,Z)$) eliminating the background where Z is more than 2250_{mm} (Algorithm 1). This filter (Threshold) means that only the obstacles in the foreground and which generate greater risks of impact in a short period of time are analyzed.

The direction of free passage is also detected in SOSS. To find out the free passage, another Threshold (line 6, Alg. 1)

filter was applied to the depth image ($DM(x,y,Z)$) excluding data where Z is greater than 3000_{mm} millimeters away from the VIP (see item 5f of Fig. 5).

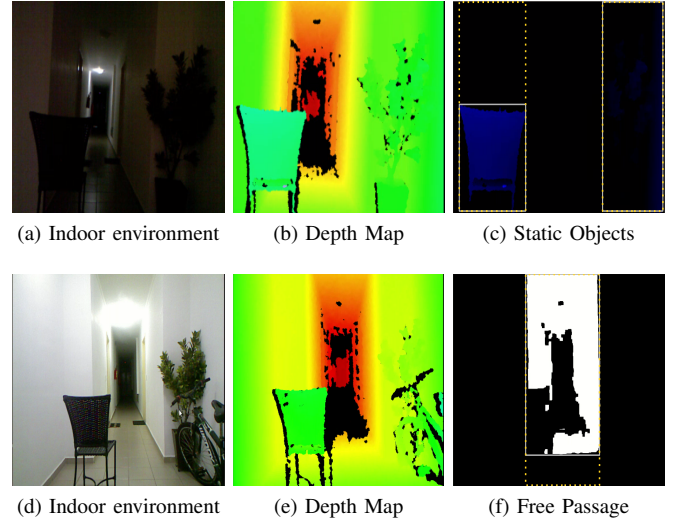


Fig. 5. Static object segmentation

Fig. 5 have the following composition: Items (5a, 5d) show the images of the indoor environment and Items (5b, 5e) are the respective depth maps; Item 5c show contour resulting from the static object segmentation method (line 7, Alg. 1); Item 5f show contour resulting from the free-passage segmentation method (line 8, Alg. 1).

It should be noted that any obstacle detected overlaps a free passage way detected in the same direction. This overlapping is necessary when the edge of the free passage invades the region containing an obstacle and this happens in the composition of the inferences shown in Figura 4.

B. Dynamic Object Segmentation Submodule

In order to provide more refined data for dynamic object analysis, a segmentation algorithm that isolates the dynamic objects from the scene (background) has been defined. The targeting method used for this purpose is Background Subtraction (BS). This technique is widely used to identify objects that are in motion;

1) *Background Subtraction*: In Fig. 6, two instants are obtained in a short time sequence and they have the following composition: Instant 1 (6a, 6b, 6c); Instant 2 (6d, 6e, 6f). Each instant is composed of three images, which show: the IR camera image (6a, 6d); the segmented dynamic object (6b, 6e); and the contour of the segmented dynamic object (6c, 6f). From these characteristics, it is possible to calculate the center of mass and the width of the dynamic object. However, this segmentation technique requires stable images.

With a defined reference system and with the objects in the environment mapped in the 3D plane, it is also possible to reconstruct the trajectory of any dynamic object. The routes are usually produced for people on the go and provide good traffic possibilities for VIP. So, choice of the paths is an important

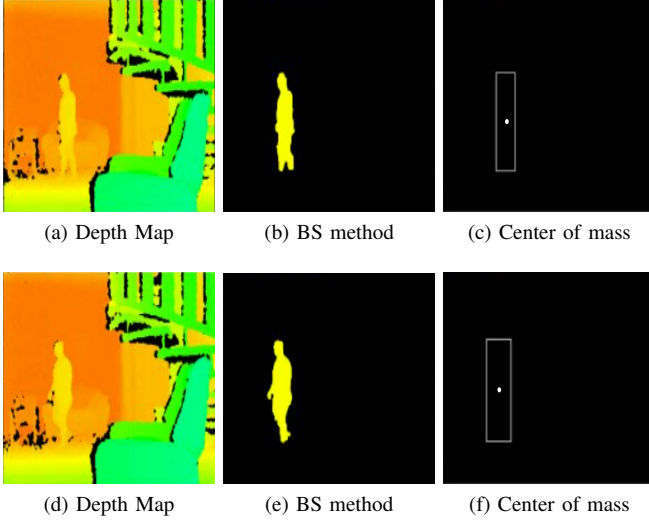


Fig. 6. Dynamic object segmentation

part of the generation of SAW in the defined context (collision risk), indicating routes with greater chances of collision-free passage. A video with different experiments can be seen in the link: (<https://youtu.be/IUSdxCCLzmk>).

C. Fusion Module

To make a comparison between classifiers, the information produced by the PEP to feed the dataset (see Fig. 4) were used. Table II shows the results of the training processes of four classifiers using cross validation with the value of 10 Folds. This table consists of the following features: the classifiers used; the Correctly Classified Instances (CCI) for each classifier; the average Precision (avg Pc) of each training; the Mean Absolute error (MAerror) rate. For this comparison, the following classifiers were used: Multilayer Perceptron, Bayesian Network, Naive Bayes and Decision Tree J48.

TABLE II
COMPARISON OF CLASSIFIERS WITH CROSS VALIDATION - FOLDS 10

Classifiers	CCI	avg Pc	MAerror
Multilayer Perceptron	94.2%	94.6%	7.41%
Bayesian Network	64.4%	68.6%	18.8%
Naive Bayes	97.1%	97.2 %	3.04%
J48(Decision Tree)	65.3%	72.7%	25.7%

V. RESULTS AND DISCUSSION

The following tables present the results of some inferences produced for the context of collision risks according to the method proposed in Fig. 4. Table III presents 20 inferences applied to ascertain the quality of the learning models generated with different classifiers. It presents data similar to those used to create a dataset of Training, in Table I.

In line 3 of Table III, the absence of both SO and DO means there is a free passage. This means that the collision risk in this direction is low. In line 5, a DO was detected at a longer

TABLE III
INFERENCES - SUPPLIED TEST SET

Example	SO _{mm}	DO _{mm}	FP _{mm}	Paths _{mm}	class
1	?	4420	3015	2015	low
2	?	?	3150	915	low
3	?	?	3315	?	low
4	?	4300	3250	?	low
5	?	4870	3085	1315	low
6	2110	?	?	1270	moderate
7	?	2855	?	?	moderate
8	2050	2850	?	1000	moderate
9	?	2710	?	910	moderate
10	2020	2680	?	1000	moderate
11	1155	?	?	705	high
12	?	1615	?	?	high
13	830	1510	?	710	high
14	1210	1645	?	630	high
15	1215	?	?	795	high
16	1005	1460	?	1520	veryhigh
17	?	1010	?	1505	veryhigh
18	750	?	?	?	veryhigh
19	?	850	?	980	veryhigh
20	?	770	?	1065	veryhigh

distance (4870_{mm}), not generating too much risk, and also, informs that there was a detection of a free passage greater than 3000_{mm} (distance from the VIP) and a very close path (1315_{mm}).

These data suggest that the VIP has a low collision risk in the next few moments following that direction. In line 8, an SO (2050_{mm}) and a DO (2850_{mm}) were detected, but there is a previously followed DO path very close. So, here, the collision risk is moderate. In the case of line 13, even if a path (730_{mm}) has already been detected, a SO (830_{mm}) has also been detected very close, making the risk high. On line 19, the risk becomes very high because the DO is close and the path detected is behind it; that is, the path indicates a good area for moving into but it is behind the DO. Table IV shows the results of the inferences corresponding to Table III.

TABLE IV
RESULTS OF THE INFERENCES

Classifiers	CCI	avg Pc	MAerror
Multilayer Perceptron	100%	100%	0.04%
Bayes Net	80%	84.4%	13.3%
Naive Bayes	100%	100%	0.01%
J48(Decision Tree)	70%	75.6%	25.1%

Table IV allowed us to conclude that the Naive Bayes and Multilayer Perceptron classifiers provided safer decision making when compared to Bayes Net and Decision Tree J48. This is due to the wide variation that can occur in the data that make up the inferences. In training models, where value ranges such as the presence of obstacles between 1000_{mm} and 1500_{mm} are limited, the Bayes Net and Decision Tree J48 increase the number of correctly sorted instances.

The results shown in Table IV are also important to confirm the reliability of the architecture. For even with some classification techniques not reaching 100% CCI, it is feasible to

feed the database in a supervised way and to improve the learning model. In addition, a safe direction can be found because the results of all inferences influence the indicated direction. As shown in Fig. 4, 16 inferences are made for the SAS-VIP to provide the final direction. The final direction must be composed of at least three consecutive directions, due to the width of the VIP. Therefore, if one direction does not have a CCI inference, the joint result of the other inferences (related to neighboring directions) can minimize this problem.

VI. CONCLUSION

The SAW model has been applied in a variety of projects. However it has not really been explored in support systems for navigation and sensory analysis by VIPs. The level of its use decreases substantially when the data fusion reaches the levels of comprehension and projection. Many projects developed to support VIP locomotion indicate the existence of obstacles, their position, distance and alternative routes to follow. However, few have applied the prediction of impacts based on the comprehension of contexts.

This project developed an architecture that provides a set of feature extractors to enable the perception and comprehension of the environment. The prediction of impacts is resulting from a set of inferences made in different directions. The results of the Vision techniques were validated by means of the fusion module. Without the availability of these characteristics, inferences could not be made.

In this way, this project has made important contributions to the development of navigation systems for VIP aiming to predict and avoid collisions. Among these contributions are: the development of an architecture to analyze the scene, the distance and position of static and dynamic objects; dynamic object path analysis; the conversion of the obstacle positions in the 2D plane to the 3D plane; indication of the direction that has the least collision risk.

VII. ACKNOWLEDGMENT

We are grateful to the Brazilian funding agency FAPESP – Project No. 2017/26421-3. Our thanks go as well to UFSCar-DC, and IFSP, SP, Brazil.

REFERENCES

- [1] S. P. Mariotti, "World health organization (who)– visual impairment and blindness," <http://www.who.int/mediacentre/factsheets/fs282/en/>, Tech. Rep., Aug 2014.
- [2] K. Y. Chan, U. Engelke, and N. Abhayasinghe, "An edge detection framework conjoining with {IMU} data for assisting indoor navigation of visually impaired persons," *Expert Systems with Applications*, vol. 67, pp. 272 – 284, 2017.
- [3] M. L. Mekhalfi, F. Melgani, A. Zeggada, F. G. De Natale, M. A.-M. Salem, and A. Khamis, "Recovering the sight to blind people in indoor environments with smart technologies," *Expert Syst. Appl.*, vol. 46, no. C, pp. 129–138, Mar. 2016.
- [4] C. Tsimmpas, A. Rompas, O. Fokou, and D. Koutsouris, "An indoor navigation system for visually impaired and elderly people based on radio frequency identification (rfid)," *Information Sciences*, vol. 320, pp. 288 – 305, 2015.
- [5] S. Mascetti, D. Ahmetovic, A. Gerino, C. Bernareggi, M. Busso, and A. Rizzi, "Robust traffic lights detection on mobile devices for pedestrians with visual impairment," *Computer Vision and Image Understanding*, vol. 148, pp. 123 – 135, 2016, special issue on Assistive Computer Vision and Robotics -.
- [6] N. Bourbakis, S. K. Makrogiannis, and D. Dakopoulos, "A system-prototype representing 3d space via alternative-sensing for visually impaired navigation," *IEEE Sensors Journal*, vol. 13, no. 7, pp. 2535–2547, July 2013.
- [7] A. Aladren, G. Lopez-Nicolas, L. Puig, and J. J. Guerrero, "Navigation assistance for the visually impaired using rgb-d sensor with range expansion," *IEEE Systems Journal*, vol. PP, no. 99, pp. 1–11, 2014.
- [8] J. Xiao, S. L. Joseph, X. Zhang, B. Li, X. Li, and J. Zhang, "An assistive navigation framework for the visually impaired," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 5, pp. 635–640, Oct 2015.
- [9] Y. Zhu, R. Y. Shtykh, and Q. Jin, "A human-centric framework for context-aware flowable services in cloud computing environments," *Information Sciences*, vol. 257, pp. 231 – 247, 2014.
- [10] M. Liggins, D. Hall, and J. Llinas, *Handbook of Multisensor Data Fusion: Theory and Practice, Second Edition*, ser. Electrical Engineering & Applied Signal Processing Series. CRC Press, 2008.
- [11] M. Endsley, B. Bolte, and D. Jones, *Designing for Situation Awareness: An Approach to User-Centered Design*. Taylor & Francis, 2003.
- [12] H. Jabnoun, F. Benzarti, and H. Amiri, "Visual substitution system for blind people based on sift description," in *Soft Computing and Pattern Recognition (SoCPaR), 2014 6th International Conference of*, Aug 2014, pp. 300–305.
- [13] S. C. Pei and Y. Y. Wang, "Census-based vision for auditory depth images and speech navigation of visually impaired users," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 4, pp. 1883–1890, November 2011.
- [14] Y. Tian, W. R. Hamel, and J. Tan, "Accurate human navigation using wearable monocular visual and inertial sensors," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 1, pp. 203–213, Jan 2014.
- [15] L. T.-L. V. N. Pham, Huy-Hieu, "Real-time obstacle detection system in indoor environment for the visually impaired using microsoft kinect sensor," 2016.
- [16] C. K. C. A. F. Kanwal Nadia, Bostanci Erkan, "A navigation system for the visually impaired: A fusion of vision and depth sensor," *Applied Bionics and Biomechanics*.
- [17] M. L. Mekhalfi, F. Melgani, Y. Bazi, and N. Alajlan, "Toward an assisted indoor scene perception for blind people with image multilabeling strategies," *Expert Syst. Appl.*, vol. 42, no. 6, pp. 2907–2918, Apr. 2015.
- [18] D. Ahmetovic, "Smartphone-assisted mobility in urban environments for visually impaired users through computer vision and sensor fusion," in *Mobile Data Management (MDM), 2013 IEEE 14th International Conference on*, vol. 2, June 2013, pp. 15–18.
- [19] P. Costa, H. Fernandes, P. Martins, J. Barroso, and L. J. Hadjileontiadis, "Obstacle detection using stereo imaging to assist the navigation of visually impaired people," *Procedia Computer Science*, vol. 14, pp. 83 – 93, 2012. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1877050912007727>
- [20] C. K. Lakde and P. S. Prasad, "Navigation system for visually impaired people," in *2015 International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC)*, April 2015, pp. 0093–0098.
- [21] R. Ivanov, "Indoor navigation system for visually impaired," in *Proceedings of the 11th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing on International Conference on Computer Systems and Technologies*, ser. CompSysTech '10. New York, NY, USA: ACM, 2010, pp. 143–149. [Online]. Available: <http://doi.acm.org/10.1145/1839379.1839405>
- [22] S. Pundlik, M. Tomasi, and G. Luo, "Collision detection for visually impaired from a body-mounted camera," in *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, June 2013, pp. 41–47.
- [23] S. L. Joseph, J. Xiao, B. Chawda, K. Narang, and P. Janarthanam, "A blind user-centered navigation system with crowdsourced situation awareness," in *Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2014 IEEE 4th Annual International Conference on*, June 2014, pp. 186–191.
- [24] N. H. Cordeiro, A. M. B. Dourado, G. da S. Quirino, and E. C. Pedrino, "A data fusion architecture proposal for visually impaired people," in *29th SIBGRAPI Conference on Graphics, Patterns and Images, SIBGRAPI 2016, Sao Paulo, Brazil, October 4-7, 2016*, 2016, pp. 158–165. [Online]. Available: <https://doi.org/10.1109/SIBGRAPI.2016.030>
- [25] G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media, 2008.