Qualitative Modeling of Indoor Environments from Visual Landmarks and Range Data

Jean-Bernard Hayet   Claudia Esteves*   Michel Devy   Frdric Lerasle

Laboratoire d'Analyse et d'Architecture des Systmes - LAAS-CNRS
7, avenue du Colonel Roche, 31077 Toulouse Cedex 04, France
e-mail: {jbhayet, cesteves, michel, lerasle}@laas.fr

Abstract

This article describes the integration in a complete navigation system of an environment modeling method based on a Generalized Voronoi Graph (GVG), relying on laser data, on the one hand, and of a localization method based on monocular vision landmark learning and recognition framework, on the other hand. Such a system is intended to work in structured environments. It is shown that the two corresponding modules — laser GVG construction and visual landmarks learning and recognition — can cooperate to complete each other, as image processing can be enhanced by some structural knowledge about the scene, whereas the GVG is annotated, even as far as its edges are concerned, by qualitative visual information.

1 Introduction

Mobile robot navigation can be considered as the art to overcome the inaccuracy of internal sensors and to take advantage of exteroceptive sensors like cameras, sonars or laser range finders to allow the robot move and act in its environment. Many strategies have already been proposed, some based on explicit localization of the robot with respect to the environment, others only relying on relative localization with respect to some interesting objects, landmarks, perceived by the robot. The topology of this set of landmarks is generally embedded in a graph. The work presented in this paper has been done in the latter framework and designed for a service robot moving in an office environment composed of a network of corridors and open spaces.

We have already presented a preliminary work in [8] using the ultrasonic-based Generalized Voronoi Graph (GVG) representation proposed by H.Choset [2]. This representation is a topological graph describing the paths on which the robot must navigate; in this approach, nodes are associated to "distinctive places", where "distinctiveness" is determined according to the US sensors. In this case, it corresponds to the discontinuities of the GVG edges, i.e. "meet points", associated to intersections between corridors or to crossings (doors) towards open spaces (rooms, hallways...). The graph edges correspond to paths in corridors or in open spaces. To overcome the classical self-localization problem resulting from US data ambiguity, we annotated each node with visual landmarks, planar, quadrangular objects (e.g. doors, windows, posters) that were automatically discovered, learned *around the meet points only*. Landmark intrinsic representations *independent from the viewpoint* were used and were shown in [5] to be stable with respect to illumination, scale changes and small occlusions.

In order to better validate an hypothesis about a node identification and to make the incremental construction more robust, we worked in two directions : (1) change the range sensor from the ultrasonic sensors belt to a *laser range finder (horizontal scanning)* and (2) annotate not only the GVG nodes *but also its edges* to maintain a qualitative position along an edge. Some authors proposed methods to deal with the incremental construction of a GVG representation using laser data. In [10], the GVG was explicitly built, coping with a lot of geometrical situations that made the method slow and unreliable. In [9], an implicit modeling strategy was proposed, using the sensor servoing, namely the task function formalism, to keep on the GVG or to detect meet points. The authors used only laser data, so that this very efficient method could have some problems in very ambiguous situations, like a regular network of corridors.

Qualitative spatial reasoning implies to work on some

*The stay of Claudia Esteves at LAAS-CNRS in France, is funded by the French-Mexican lab in Computer Science.

space notions without using any representation or reasoning method requiring numeric or quantitative descriptions. Qualitative information covers *topology* — connectivity, topological relationships — *orientation*, and *order*. In [3], the available approaches to model *topological* relationships were reviewed, and some new ones were proposed. *Orientation* and *order* relationships are also widely used : in [4] the notion of intrinsic orientation is underlined, i.e. the orientation relative to the robot current position on its trajectory.

Nevertheless, in order to achieve robust navigation, it is desirable to forget the pure "qualitative" notions so that the robot could benefit from all the possible data it could use and combine metric and topological levels of information. As an example, Kuipers [7] introduced the notion of *Spatial Semantic Hierarchy* that included, among the others, two levels for topological and metric information.

The sections 2 and 3 present the modules devoted respectively to the GVG construction from laser data and to the landmarks detection from visual input. In section 4, we introduce the compound environment representation and finally the section 5 sums up this work and opens a discussion for our future works.

2 Building a GVG from laser data

The Generalized Voronoi Graph representation associates the set of points equidistant from at least two obstacles to its edges and *meetpoints* — points equidistant to at least three obstacles — to its nodes. The latter ones are salient features in the environment, distinctive places, such as corridor intersections, crossings to open spaces (rooms or hallways) and corridor ends. The incremental construction of the GVG does not only provide a natural way to capture the topology of the environment free space but also greatly reduces the error accumulation due to odometry by observing the change of local coordinates. The GVG construction consists in going over every possible path in a corridor-based environment, memorizing the path connections in the GVG and learning visual landmarks at the nodes and along the edges, as presented in section 3. Note that the two traditional tasks, exploration and navigation, have no clear boundary here, as they are performed at the same time.

The robot can be controlled to navigate along a GVG edge by keeping equidistant to the two closest obstacles which are mainly the two walls on the corridor. The inputs of this control law are the distance and orientation to the GVG edge computed from the segment information provided from the laser range finder. A prediction and correction steps are merged to obtain a smooth path. Detected laser segments provide a representation (figure 1) that is required either to keep on the GVG ((a) and (f)), to detect a meet point ((b) and (c)) or to detect an obstacle or a dead-end ((d) and (e)).

Because the laser range finder sensor we are using has a limited angle, a meetpoint detection approach by watching for an abrupt change in the direction of the gradients to the two closest obstacles as proposed in [2] becomes unsuitable. In order to detect and move to a meetpoint, our approach relies on the observation of filtered and segmented range data. Two main *events* on the tracked corridor can be identified. One, when two segments belonging to the same wall on the corridor are disconnected by a length superior to a given threshold (*discontinuity*) and the other at the end of a wall (*end*). Such events are closely related to the nature of the other obstacles found on the same scan as they can be produced due to occlusions, open doors or new paths.

A model of the approaching meetpoints *mp* can be determined according to the configuration of these events and the nature of the found obstacles before the robot actually gets there, typically at about 5 meters from the meetpoint. The underlying hypothesis generation-verification scheme relies on the a priori knowledge of models presented on the figure 1. The following lines sum up our strategy :

```
0: Search for two major line segments within the segmented data → wall1 and wall2.
1: Access to GVG by gradient ascent. mp = ∅
2: Main loop.
   while (remaining_paths ≠ ∅) do
     if (closest(mp) not reached) then
       track wall1 and wall2
       check for events
     else
       confirm closest(mp)
       update graph and follow exploration
     end if
     update mp
   end while
```

As an illustration, figure 2 shows the robot getting onto the GVG (1), following it while generating hypothesis (2) and reaching a meet point (3). Note that obstacles inside the corridor are easily detected, so the robot performs avoiding strategy (figure 1,(f)) or adds a dead-end (figure 1,(e)), according to the obstacles relative size and orientation.

3 Visual landmarks detection

With corridor-like environments, extracting vanishing points and skyline is relatively easy. We show

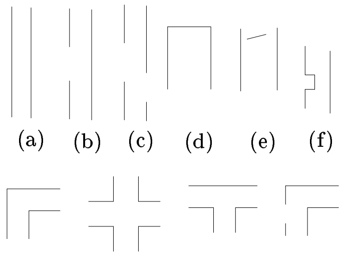

Figure 1: *Corridor and meetpoints configurations*

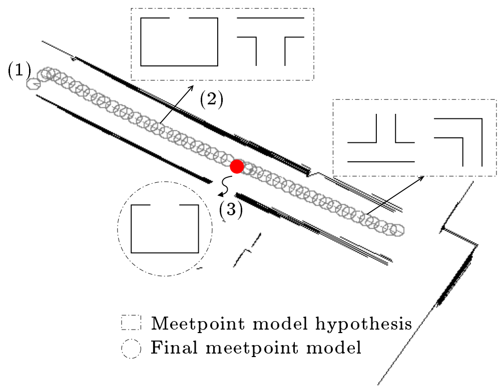

Figure 2: *GVG incremental construction*

how we can take this into account to improve our landmark detection strategy.

3.1 Vision in corridor-based environments

With some simple assumptions about the environment, well-adapted visual functions can be proposed to help the robot navigate. We focused our previous work on the use of rectangular visual landmarks. We suppose for the moment that the camera intrinsic parameters matrix K is known :

$$K = \begin{pmatrix} \alpha_i & 0 & i_0 & 0 \\ 0 & \alpha_j & j_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

The use of the *vanishing points* information inside the image processing steps seems inevitable in this case. Let i and j be the image coordinates. Let ϕ be the platform tilt angle and θ the horizontal angle between the camera optical axis and the corridor direction. As we illustrate it on figure 3 the robot planar motion constraint and the camera platform movements restricted to pan and tilt motions make the skyline $i = i_s$ and *vertical* vanishing point $p_v = (i_v, j_v, t_v)$ (in homogeneous coordinates) be known. The platform and camera internal parameters K are read to have :

$$\begin{cases} i_s = i_0 - \alpha_i \tan(\phi) \\ i_v = i_0 \tan(\phi) + \alpha_i \\ j_v = j_0 \tan(\phi) \\ t_v = \tan(\phi) \end{cases}$$

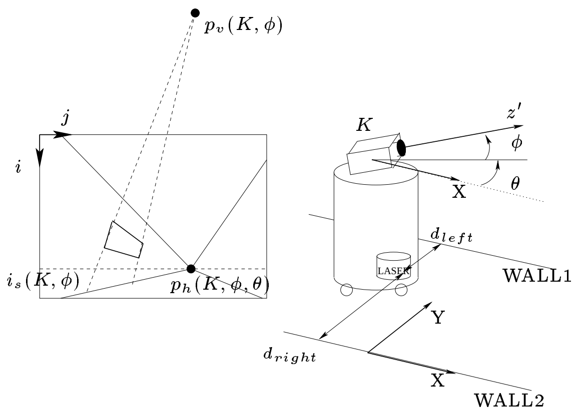

Figure 3: *Spatial configuration in a corridor*

Moreover, the knowledge of the skyline is very useful to perform, on a second step, a quick search of the *horizontal vanishing point* p_h, reduced to one dimensional problem *along the skyline*. This approach is a classical one, but requires (1) an image segmentation into edge segments and (2) a Hough-like transform. It is not very convenient for on-line processing but may be useful, as it will be explained hereafter.

3.2 Laser/camera transformation

A key problem to use laser data in our image processing functions is to have a good estimate of the transformation T_{sc} between the two sensors. Let $(\alpha_{sc}, \beta_{sc}, \gamma_{sc}, tx_{sc}, ty_{sc}, tz_{sc})^T$ be the transformation parameters and T_{sc} the corresponding 4×4 matrix. Physical measuring allows to have a first approximation of T_{sc}. A reasonable hypothesis is that γ_{sc}, resulting from the two roll angles, is close to zero. The other parameters have to be found in a preliminary *calibration phase*.

An interesting method to calibrate the T_{sc} transform consists in *decoupling the angles and translations* parameters thanks to the infinite points. Indeed, let be some corridor images, from both laser and camera. From visual segments, we can apply the Hough transform-based search we mentioned above to get a visually detected horizontal vanishing point $p_h^v = (i_h^v, j_h^v, 1)$ corresponding to the corridor. We can also re-project the infinite point from laser data into a point $p_h^r = (i_h^r, j_h^r)$ depending on T_{sc}. Defining :

$$\begin{cases} f(i) = \arctan(\frac{i_0 - i}{\alpha_i}) \\ g(i,j) = \arctan(\cos(f(i))\frac{j-j_0}{\alpha_j}) \end{cases}$$

α_{sc} and β_{sc} are computed by minimizing :

$$\begin{cases} \alpha_{sc} = \arg\min_\alpha (i_h^v - (i_0 - \alpha_i \tan(f(i_h^r) + \alpha)))) \\ \beta_{sc} = \arg\min_\beta (j_h^v - (j_0 + \frac{\alpha_j \tan(g(i_h^v, j_h^r)+\beta)}{\cos(f(i_h^v))})) \end{cases}$$

Finally, the angle errors we get are about 0.02 radians, as seen on figure 4. The histogram represents the angular error on θ corresponding to the shift between p_h^v and p_h^r, for the computed α_{sc}, β_{sc}.

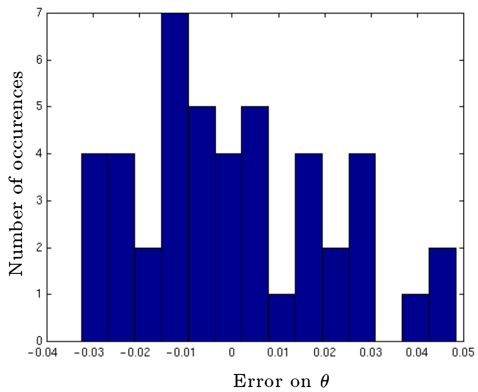

Figure 4: *Error on θ over a test sequence*

At this point, let's define the **corridor lines** by the four parallel straight lines d_k, $k = 1..4$ defining the corridor 3D model. Then, matching the corridor lines projections **detected** onto the image with the model-based **re-projected** ones allows to find the translation parameters. However, the needed precision on these parameters is much less important than the one on the angles, as they will only have influence on the corridor lines projections, not on the directions of detected primitives.

3.3 Projection of laser data

Laser data from the GVG module may provide some useful information to enhance our visual functions. Indeed, we have proposed in [1] a simple method to detect planar, quadrangular landmarks lying on a vertical wall, posters for example. One of the key features of this system was that salient zones detection and segmentation was partially done in a *1D* image resulting from an *averaging procedure over the whole image* along *vertical* direction only. No a priori information about the scene or the robot was used.

However, when the robot enters a corridor-like part of the environment, laser data can provide a reliable estimate of the robot direction along the corridor, so we can get the θ angle from figure 3 and the *horizontal* vanishing point $(i_h, v_h, 1)$:

$$p_h = (i_h, j_h, 1)^T = (i_s, j_0 - \alpha_j \frac{\tan(\theta)}{\cos(\phi)}, 1)^T$$

When the GVG module computes the distances d_{left} and d_{right} to the wall (see figure 3), we can get the four projected corridor lines d_k. We know that they go through point p_h so that for each d_k we only need one more point projection. We can take the intersections of the camera horizontal axis with the wall. For the right wall bottom line d_1, for instance :

$$d_1 = (p_h, KT_\phi T_{sc} \begin{pmatrix} -d_{right}\sin(\theta) \\ d_{right}\cos(\theta) \\ -H_s \\ 1 \end{pmatrix})$$

H is a height arbitrarily chosen, H_s the laser height from the floor.

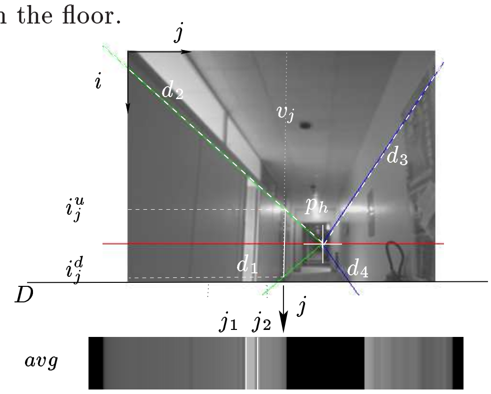

Figure 5: *How laser enhances image processing*

3.4 Detection of salient quadrangles

From the four d_k half straight lines, we can perform an efficient search : let us consider the j values along the horizontal line D, at the bottom of the image, as illustrated in figure 5, with j varying from $j = j_{min}$ to $j = j_{max}$. The equation of D is $i = i_{max}$.

In each j we can then define a direction v_j passing through p_v and two bounds i_j^u and i_j^d on this direction corresponding to the projections of the areas of the lateral walls. The averaging procedure can be done *on this segment only* to get a 1D image *avg* as in figure 5. *avg* is processed to detect salient transitions on D points j_k. These transitions are treated separately to isolate corresponding vertical segments in the image.

From all the points we detailed before, we can now present the detection algorithm as follows :

0: Gets *attitude_data* (ϕ)
1: Computes p_v from *attitude_data*
2:
  if (ReadCorridorInfoFromGvg()=OK) **then**
    gets θ and computes p_h and d_k, $k = 1..4$
    set vertical averaging bounds from d_k
  else
    set vertical averaging bounds to default
  end if
3: Averages $\rightarrow$ image *avg* along lines v_j
4: Detects in *avg* transitions points j_k
5:
  for all k **do**
    Detects transitions along v_{j_k}.
    Segmentation and RANSAC estimation.
  end for
6: Matches vertical segments together by relaxation.
7: Closes all matched pairs.

The closure procedure, already described in [1], is based on a RANSAC function. We adapted it to take the vanishing point into account. The closure may be *total*, when both lower and upper vertices have been isolated, as the left quadrangle in figure 6; it may be *partial* when, as the right example from the same figure, only one horizontal edge has been found.

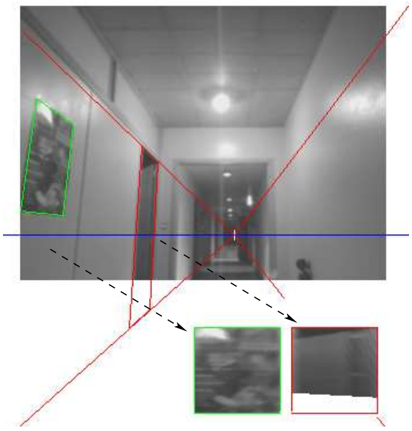

Figure 6: *Results of landmark detection in a typical corridor environment*

The landmark representation is computed from the "iconification" of detected quadrangular landmarks, by applying an homography H on the original image to a 75×75 square SQ. These icons are shown on figure 6. More details can be found in [5].

3.5 Case of partially detected landmarks

The *partially* detected landmarks compose the majority of detected landmarks in indoor environments : doors, cupboards are very frequent in office environments. As we have only three available lines and we need four lines to perform the representation construction, we propose to use the corridor lines to complete them, and we call "door-like" this kind of landmark.

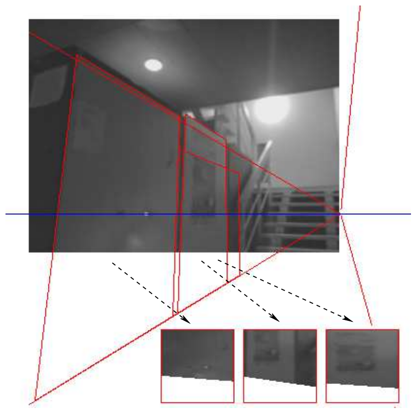

Figure 7: *Partially detected "door-like" landmarks*

Figure 7 illustrates this process on a set of cupboards, where landmarks models are defined from H between the three detected lines and d_1, on the one hand, and the previously defined square SQ, on the other hand.

4 Integrating visual landmarks into the environment representation

Once landmarks have been detected, we have to integrate them into the graph-based representation of the environment. The GVG approach can also embed this kind of information. We saw in [8] that nodes could be annotated with visual landmarks. In this work, we also try to enrich the graph edges with visual information.

There are different levels of information we have to process from visual landmarks :

- *intrinsic data* embedded in the landmark.
- *orientation* relatively to the edge
- *topological and order relationships* with other landmarks

The intrinsic data are extracted from the iconified views, as described in [5]. This appearance representation is robust to illumination, viewpoint and scale changes, so that we are able to recognize the same landmark at different points in the corridor.

Orientation gives the position of the landmark in one of the corridor sides : left/right. Last, topological and order information represent the relative relationships between landmarks, if available : relative positioning along the wall, relationships of inclusion/intersection/disjunction. As in figure 7, landmarks do not necessarily correspond to physically distinct objects.

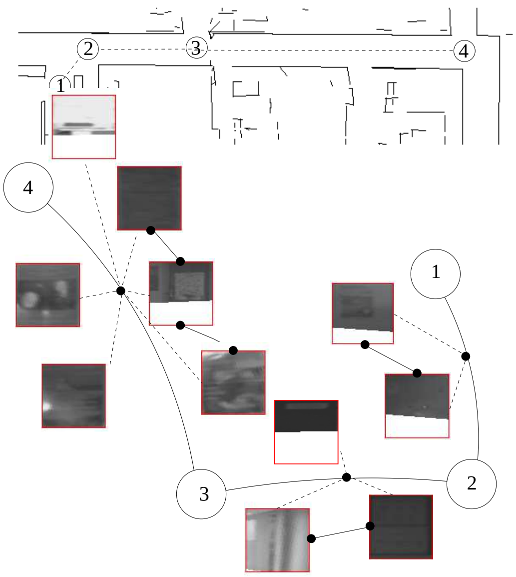

Figure 8: *A graph for visual and laser information*

Figure 8 shows an example of such a graph for the portion of the metric space represented on the upper part. Dashed lines represent links from landmarks to edges, solid lines the existing inter-landmarks topological relationships. Note that not all the connections between landmarks from the same walls have been defined.

Maintaining qualitative knowledge is still part of our current work. We base our approach on [6], where orientation and topology are taken into account simultaneously.

5 Conclusion and future works

This paper has presented the integration of two topological based representations required for the navigation of a mobile robot in an office environment. We take advantage both from the GVG model, suitable to represent a network of corridors, and from a landmark-based topological map to provide a qualitative localization of the robot with respect to the nodes and the edges of the GVG.

The GVG construction relies on laser data, more accurate and less noisy than the ultrasonic data we used in a preliminary work [8]. The landmark learning and recognition method has been improved to become more robust and more reactive. In the corridors, this module takes profit of the laser data to focus the landmark detection procedure only on the lateral walls.

In our lab, the environment is more complex than a simple network of orthogonal corridors, so that the GVG approach is very useful, but vision is mandatory to guarantee a good recognition of the nodes. We are currently trying to get more significant experimental results, including the crossing of open spaces like large hallways, in which the topology cannot be reliably described by a GVG. For such places, we intend to associate edges to visually-controlled actions (for example, *goto landmark* or *follow the line on the wall*, ...). We also intend to consider more types of visual landmarks: vertical edges, lines on the ground or on the walls, quadrangular and planar objects located on the ceiling . . . so that the robot could always locate itself.

References

[1] V. Ayala, J.B. Hayet, M. Devy and F. Lerasle. Visual Localization of a Mobile Robot in Indoor Environments using Planar Landmarks IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Takamatsu, Japan.

[2] H. Choset, K. Nagatani and A. Rizzi. Sensor Based Planning: Using a Honing Strategy and Local Map Method to Implement the Generalized Voronoi Graph. SPIE Mobile Robotics, Pittsburgh, PA, 1997.

[3] E. Clementini and P. DiFelice. A Comparison of Methods for Representing Topological Relationships. Information Sciences,3,149-178,1995.

[4] C. Freksa and K. Zimmermann. On the Utilization of Spatial Structures for Cognitively Plausible and Efficient Reasoning. IEEE Int. Conf. on Systems, Man and Cybernetics, October 1992.

[5] J.B. Hayet, F. Lerasle and M. Devy. A Visual Landmark Framework for Indoor Mobile Robot Navigation. IEEE Int. Conf. on Robotics and Automation, May 2002, pp. 3942-3947, Washington, USA.

[6] D. Hernández. Maintaining Qualitative Spatial Knowledge. European Conf. on Spatial Information Theory, Elba, 1993.

[7] B. Kuipers. The Spatial Semantic Hierarchy. Artificial Intelligence. 2000.

[8] P. Ranganathan, J.B. Hayet, M. Devy, S. Hutchinson and F. Lerasle. A Visual Landmark Framework for Indoor Mobile Robot Navigation. Int. Symp. on Intelligent Robotics Systems, Toulouse, France, 2001.

[9] A.C. Vitorino, P. Rives and J.J. Borrelly. Mobile Robot Navigation Using a Sensor-Based Control Strategy, IEEE Int. Conf. on Robotics and Automation, Seoul, Korea, May 2001.

[10] D. Van Zwynsvoorde, T. Simeon and R. Alami. Incremental Topological Modeling using Local Voronoi-like Graphs, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Takamatsu, Japan.