Dpto. de Informática e Ingeniería de Sistemas
Universidad de Zaragoza
C/ María de Luna num. 1
E-50018 Zaragoza
Spain

Internal Report: 1998-V04

Navigation from uncalibrated monocular vision

J.J. Guerrero, C. Sagüés

If you want to cite this report, please use the following reference instead:
Navigation from uncalibrated monocular vision. J.J. Guerrero, C. Sagüés, *IFAC, Intelligent Autonomous Vehicles*, pages 210-215, 1998.

NAVIGATION FROM UNCALIBRATED MONOCULAR VISION

J.J. Guerrero, C. Sagüés

Dpto. de Informática e Ingeniería de Sistemas,
C.P.S. de Ingenieros, Universidad de Zaragoza,
María de Luna 3, E-50015, Zaragoza, Spain
Phone 34-976-761940, Fax 34-976-762111
email: {jguerrer, csagues}@posta.unizar.es

Abstract:
In this paper we propose a method to correct the heading of an indoor mobile robot using an uncalibrated monocular vision system. Neither environment map nor explicit reconstruction is obtained and no memory of the past is recorded. We extract straight edges and classify them as vertical and non-vertical. From the non-vertical lines we obtain the vanishing point to compute the robot orientation. From corresponding vertical lines in two uncalibrated images we obtain robot heading using the focus of expansion, and a qualitative depth map used to compute the commanded heading.

Keywords: Vision based navigation, uncalibrated vision, straight lines, motion and structure.

1. INTRODUCTION

Mobile autonomous robots are actually able to execute tasks indoors, where robot localization is obtained using specific landmarks. However, without this assumption, powerful perception systems to correct the motion of the robot are required. Vision is a sensor broadly used.

Many works on vision for robot navigation try to build accurate models of the scene, using very accurately calibrated systems (Ayache, 1991). These systems work, but normally are computationally expensive and conditioned by the fitting of the real scene to the model made for navigation.

In our approach we do not reconstruct the environment, and the robot moves as the vision system acquires information from the current scene. The emphasis is made on simplicity to obtain a reasonable speed with computational resources on the own robot. Only two images are used and no memory of the past is recorded.

As neither previous map nor robot heading is assumed, we must obtain structure and motion information from vision (Sagüés and Guerrero, 1995). The computation of the direction of the translation and the time to collision based on the focus of expansion (FOE) is a classical result (Burger and Bhanu, 1992), (Micheli *et al.*, 1993). In these approaches the basic visual information is usually point correspondences or optical flow measures, and accurately calibrated cameras are usually considered.

On the other hand, it has been shown that it is possible to recover information from uncalibrated cameras, based in its projective properties (Luong and Faugeras, 1996).

In this paper we use a monocular vision system without previous calibration. Odometry provides quite well the length advanced, but orientation and heading are not robust enough (Lebégue

[1] This work was partially supported by spanish CICYT projects TAP94-0390 and TAP97-0992-C02-01

and Aggarwal, 1993), and therefore they must be corrected. We extract straight edges in the images (Burns *et al.*, 1986) and classify them as vertical and non-vertical. From the non-vertical lines, the vanishing point is obtained using the Hough transform.

Here, we do not assume that the intrinsic parameters of the camera are known (pixel aspect ratio, focal lengths, coordinates of the principal point) and we do not compute the extrinsic camera calibration with respect to the mobile robot. We only suppose that the image plane is approximately vertical, and that the focal axis points approximately in the direction of advance. From these assumptions, we compute the FOE and a relative depth information using corresponding vertical lines in two images.

A summary of the paper follows. In §2 we present the control scheme including the vision module to carry out robot navigation. In §3 the extraction and matching of straight edges is presented. In §4 we explain how we obtain the commanded heading from non-vertical lines. In §5 we explain the method to compute the robot heading and the steps to collision. Finally we present some experimental results and the conclusions.

2. IMAGE BASED CONTROL SCHEME

To control robot heading, we propose a control scheme, directly in image coordinates. The robot is supposed to move forward step by step on an horizontal ground plane of an indoor environment. The robot is assumed to move in a corridor or a room with a well defined main direction and goal is to advance along the corridor, avoiding collisions. It makes a heading correction in each step based in the images of this step and the previous one.

In Fig. 1 the vision module can be seen. It takes two images as inputs. Firstly, straight lines are extracted and classified as vertical and non-vertical. From the non-vertical lines a main vanishing point (VP) is obtained in each image. On the other hand, the vertical lines in both images are matched to have corresponding lines. The direction of translation of the last step is obtained from the focus of expansion (FOE), using these vertical lines and the vanishing points. Once the FOE has been obtained the relative depth of vertical lines is evaluated in steps to collision units. This information is used to determine qualitatively an image region (FS), which corresponds with the free space in front of the robot. Therefore, the output of the vision module are: the vanishing point which can be considered the dominant direction of the scene, the FOE which corresponds

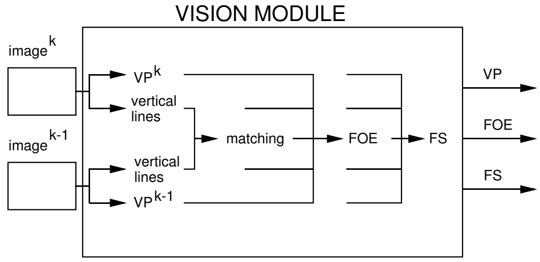

Fig. 1. Outline of the vision module.

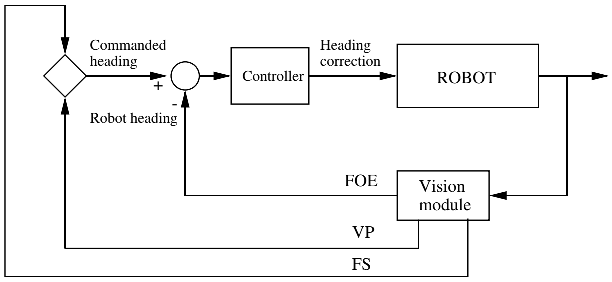

Fig. 2. Outline of the heading control scheme based on vision.

with the robot heading in the last step, and an image region of free space in front of the robot. All of these measures are x positions along the horizontal direction, and they are expressed in pixel coordinates in the last image.

The image based control scheme is outlined in Fig. 2. It corrects in function of the difference between a commanded robot heading and the robot heading (measured from the FOE). The goal is to have a robot heading equal to the commanded heading, making a heading correction in each step. The commanded heading is obtained from the vanishing point when there is enough free space in front of the robot, or from the center of the free space region when the relative depth in front of the robot is small. A simple controller allows to correct the heading in function of the heading error (measured in the image). At the moment a proportional controller has been implemented, but more complex controllers are not discarded.

3. LINE FEATURES. EXTRACTION AND MATCHING

As was mentioned above, our approach takes straight edges in the image as key features. They are extracted with our version of the method proposed by Burns (Burns *et al.*, 1986). This extractor provides not only the geometric representation of the projected line, but also some attributes related with the brightness of the edge (contrast, average gray level, steepness). These bright attributes provide information very useful to identify the edge.

Once straight edges are extracted in the image, they are classified as vertical and non-vertical. To do that we assume the image plane is approximately vertical, and therefore vertical lines appear parallel and vertical in the image.

After that, the vertical lines in two images are matched. We determine correspondences between linear segments in two images without assumptions about the motion or the structure of the scene. We match features with the nearest neighbor in the second image using a statistical distance similar to the proposed in (Deriche and Faugeras, 1990). But we use not only geometric parameters but also brightness attributes supplied by the contour extractor (Guerrero and Martínez, 1995).

The image segment representation is composed of 5 parameters (x, y the midpoint coordinates, l the segment length, agl the average grey level, c the contrast). We select the corresponding in function of the difference between the measured values of those parameters in both images. Normalizing the error with its covariance, the similitude function (sf) is

$$sf = \frac{(x_1 - x_2)^2}{\sigma_x^2} + \frac{(y_1 - y_2)^2}{\sigma_y^2} + \frac{(l_1 - l_2)^2}{\sigma_l^2} + \frac{(agl_1 - agl_2)^2}{\sigma_a^2 gl} + \frac{(c_1 - c_2)^2}{\sigma_c^2}$$

4. DOMINANT SCENE DIRECTION

As was mentioned above, the robot is assumed to move in a corridor or a room with a well defined main direction which is the main direction to advance. In these conditions, the orientation of the robot can be computed from vanishing points.

We compute the vanishing point from non-vertical lines appearing in the image. Vanishing point is defined as the point where the projection in the image of parallel 3D lines converge. With the non-vertical lines, two vanishing points appear. One is the corresponding to the main direction of the environment and therefore corresponds to the lines appearing in the direction of motion. The other is the corresponding to the lines which are approximately perpendicular to the direction of motion, and its computation is ill-conditioned. We have used the first one. When the scene depth in front of the camera is high, we compute commanded heading from this vanishing point.

From non-vertical lines the vanishing point is obtained using the Hough transform. To apply the Hough transform, we partition the line of the horizon, and we obtain the intersections of the non-vertical lines with the line of the horizon. The interval where more intersections are found gives the vanishing point.

As the proposed method does not require camera calibration, and the camera is not totally vertical, the line of the horizon will not be the line of the center of the image. Therefore we must look for the vanishing point which may be on any line of the image plane. The process previously explained is repeated with a set of horizontal lines around the center of the image. At the end, we select as the line of the horizon that which has the best defined vanishing point.

5. ROBOT HEADING AND STEPS TO COLLISION FROM LINES WITH AN UNCALIBRATED CAMERA

Using lines in two images the general problem of structure and motion cannot be solved (Huang and Netravali, 1994). A line in each image provides two parameters, but four parameters are also needed to determine its 3D position. The same happens with a vertical image plane and using only vertical lines moving on a horizontal plane. Each line provides one parameter in each image (its x coordinate), but two parameters are needed to locate the vertical line on the ground plane. Therefore, there are not enough information to compute motion and structure from lines in two images. However, in this reasoning, constraints like the visibility constraint were not considered.

5.1 *Computation of the robot heading from the FOE*

Using the visibility constraint (the scene observed is always in front of the camera), and assuming pure translational motion or general rigid motion with a bounded rotation, it is possible to solve, at least qualitatively, the structure and motion problem from normal flow information (Aloimonos and Duric, 1994), (Guerrero, 1996). This idea can be extended to corresponding lines in two images.

When the relative motion between the camera and the scene is a pure translation, the projected motion field has one singular point which is a focus of expansion or contraction. The image trajectories of every point are linear and they intersect at the FOE, which corresponds with the projection in the image of the direction of the translation.

We obtain the FOE from corresponding lines in two images using a voting scheme that exploits the visibility constraint. As the observed lines must be in front of the camera and the robot makes a forward motion, all the image velocities

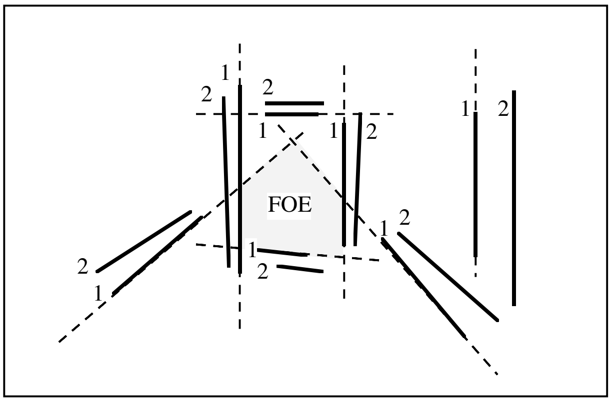

Fig. 3. In this figure corresponding lines in two images are shown (1 and 2 indicate the image number). Every line constraints the FOE to belong to a half plane. The intersection of all these half planes provides a solution region.

must expand from the FOE. When a line moves from the left to the right, every point belonging to the half plane on the left of the line can be considered as a possible FOE. On the other hand, the FOE will be on the right of a line when this line moves from the right to the left. A voting scheme based on the Hough transform is used to obtain the FOE. Thus, each line votes to all the image points compatible with the sign of its projected motion. After the voting process the image region having a higher number of votes is selected (Fig. 3). This method provides an image region as possible FOE, whose size depends on the number and distribution of the lines used. It is interesting to have as many matched lines as possible, in order to obtain a narrow FOE.

In this work only vertical lines and planar motion is assumed, which simplifies the computation. Therefore the accumulator of votes has one dimension and only the x position of the FOE is computed. Naming x_1, x_2 the location of corresponding vertical lines in first and second images respectively, the algorithm can be outlined as follows:

begin FOEComputation
    for all matched lines
        if $x_2 > x_1$ vote to all $x_{ac} < x_1$
        if $x_2 < x_1$ vote to all $x_{ac} > x_1$
    end for
    search the most voted in x_{ac}
end

where x_{ac} is the accumulator array.

The method to compute the FOE just presented assumes that the camera makes null or very small rotation. As the vanishing point corresponds to a point in the infinite, its location in the image change only due to rotations but not due to translations. The effect of the relative camera rotation can be approximately corrected from vanishing points in both images. To do that, the locations of the lines in the second image are corrected before the Hough transform is applied. The proposed correction of the rotation is $(VP_2 - VP_1)$, where VP indicates the x coordinate of the vanishing point, and therefore $x_2^{corrected} = x_2 - (VP_2 - VP_1)$.

This approximation is more valid near the VP. In our system the VP is near the FOE, where the correction of rotation is more important, because the effect of camera translation is small and the sign of the projected motion can change due to a small rotation.

5.2 *Computation of the steps to collision*

Once the direction of the translation has been obtained, a relative depth map can be obtained assuming only the epipolar geometry. The computation is equivalent to the time to collision computation obtained from optical flow measures or corresponding points in calibrated cameras (Burger and Bhanu, 1992), (Micheli *et al.*, 1993). However, we obtain them from corresponding vertical lines in two uncalibrated images.

The formulation to obtain this relative depth can be achieved based on a projective invariant. Assuming that the robot advances with constant velocity, the relative depth is obtained in steps to collision units. We can see in Fig. 4 the geometry involved in this problem assuming vertical image planes and vertical extracted lines (the image is a line and the vertical edges are points). It is known that the cross-ratio of four points on a line is preserved under projective transformations (Mundy and Zisserman, 1992). Thus from four collinear points (a, b, c, d) we can define a projective invariant Cr that is,

$$Cr(a, b, c, d) = \frac{(x^c - x^a)(x^d - x^b)}{(x^c - x^b)(x^d - x^a)}$$

where $(x^c - x^a)$ is the distance between points c and a.

To derive the expression that provides the steps to collision of a observed vertical line, we define four aligned points in the second image. They are:

a) the point on the infinite,
b) the projected location of the vertical line in the second image ($x^b = x_2$),
c) the projected location of the vertical line in the first image ($x^c = x_1$),
d) the FOE location.

Thus, their cross-ratio is,

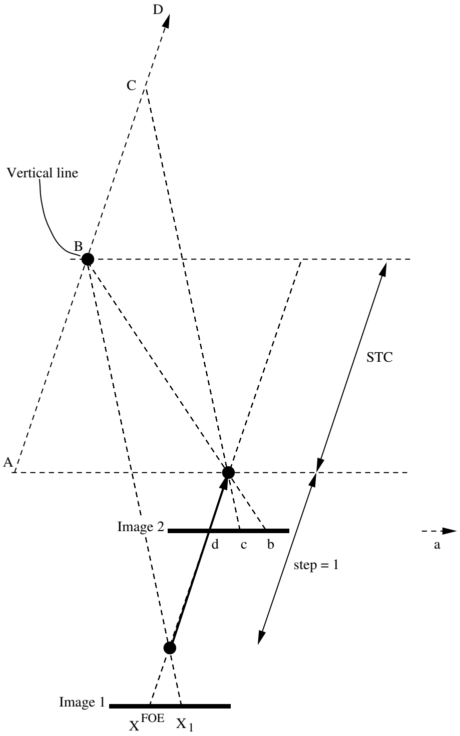

Fig. 4. Top view of the camera motion and scene assuming the camera translates.

$$\begin{aligned} Cr(a,b,c,d) &= \frac{(x^c - x^a)(x^d - x^b)}{(x^c - x^b)(x^d - x^a)} \\ = \frac{\infty\,(x^{FOE} - x_2)}{(x_1 - x_2)\,\infty} &= \frac{x^{FOE} - x_2}{x_1 - x_2} \end{aligned} \quad (1)$$

where x^{FOE} is the projection in the image of the direction of the camera translation.

These four points can be considered the projection of four points on a line on the scene feature, that is parallel to the translation (points A, B, C, D in Fig. 4), whose cross-ratio is,

$$\begin{aligned} Cr(A,B,C,D) &= \frac{(x^C - x^A)(x^D - x^B)}{(x^C - x^B)(x^D - x^A)} \\ = \frac{(1+STC)\,\infty}{1\,\infty} &= \frac{1+STC}{1} \end{aligned} \quad (2)$$

where the distance that advances the robot between two images is considered the unit, and STC is the number of steps to collision of the line.

Therefore, the cross-ratio of this four points and the cross-ratio of their perspective projection must be equal. Thus, equating (1) and (2) we arrive at,

$$STC = \frac{x_1 - x^{FOE}}{x_2 - x_1} \quad (3)$$

which allows to obtain the steps to collision of a vertical line knowing the FOE and the locations of the line in two images. In this computation it is also assumed that the camera does not rotate. When the camera rotates, the effect of this rotation must be corrected as explained in §5.1.

Fig. 5. An example of the vanishing point, obtained from non-vertical lines.

From the relative depth of several vertical lines, a qualitative free space is determined as the region where the lines have more than a given number of steps to collision. Usually the lines near the FOE have a inaccurate relative depth, because the triangulation is ill conditioned (the baseline and the projection lines are nearly parallel), and therefore they must be eliminated. When this free space is narrow or it is not centered in the image, the center of the free space is used as commanded heading.

6. EXPERIMENTAL RESULTS

Some experiments with a Labmate robot have been made. The camera is mounted on the robot, with its image plane approximately vertical. We have moved the robot on a corridor. Using the VP, the robot can go and return along the corridor many times without collision, because it has always an absolute measure of its orientation with respect to the environment. Besides that, the robot can correct its lateral position to cross a door on the corridor (using the FS and the FOE). Nothing of these tasks can be made with a priori map and planning, because of odometric drifts.

As an example, the extraction of main scene direction from lines in an image can be seen in Fig. 5. In this case, the vanishing point in the image provides the commanded heading because there is enough free space in front of it. In Fig. 6 the robot heading (FOE) has been obtained from two images. This allows to compute the free space in front of the robot (between IZQ and DER in Fig. 6). In this other case the commanded heading is obtained from the center of the free space region.

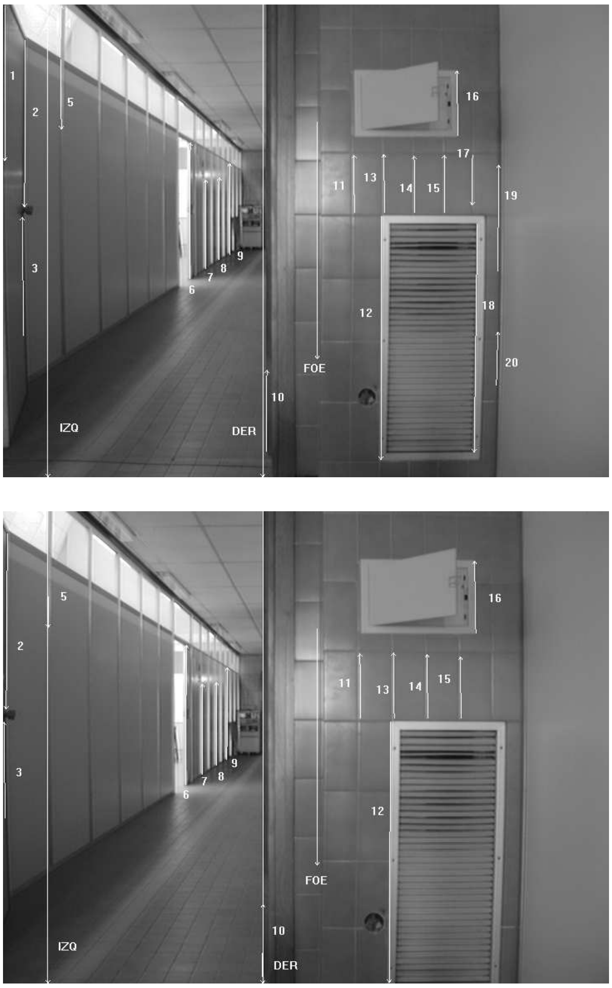

Fig. 6. From corresponding lines in two images, we obtain the robot heading (*FOE*) and the free space (IZQ/DER) to advance. We show here only the lines used to obtain the free space. To obtain the *FOE* many more corresponding lines have been used.

7. CONCLUSION

In this paper, we correct the heading of a mobile robot from information of straight edges obtained with a camera. The system uses a control scheme that extracts from the image both the robot heading and the commanded heading. Assuming the problem as planar, heading corrections are achieved in two images without calibration.

The vanishing point provides the orientation of the robot and a commanded heading to guide it. Using the epipolar and visibility constraints it is possible to compute both the robot heading and a qualitative map of steps to collision. We have proposed to compute this information from corresponding infinite lines in two uncalibrated images, and the results are good. We expect to do more experiments to reinforce this technique and to compare it with alternate approaches.

8. REFERENCES

Aloimonos, Y. and Z. Duric (1994). Estimating the heading direction using normal flow. *Int. Journal of Computer Vision* **13**(1), 33–56.

Ayache, N. (1991). *Artificial Vision for Mobile Vision. Stereo Vision and Multisensory Perception.* MIT Press. London.

Burger, W. and B. Bhanu (1992). *Qualitative Motion Understanding.* Kluwer Academic. Boston.

Burns, J.B., A.R. Hanson and E.M. Riseman (1986). Extracting straight lines. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **8**(4), 425–455.

Deriche, R. and O. Faugeras (1990). Tracking line segments. In: *First European Conference on Computer Vision.* Antibes, France. pp. 259–268.

Guerrero, J.J. (1996). Percepción de Movimiento y Estructura con Visión basada en Contornos Rectos. PhD thesis. Dpto. de Informática e Ingeniería de Sistemas, Universidad de Zaragoza. Spain.

Guerrero, J.J. and J.M. Martínez (1995). Determination of corresponding segments by tracking both geometrical and brightness information. In: *International Conference on Advanced Robotics.* Barcelona. pp. 703–709.

Huang, T.S. and A. N. Netravali (1994). Motion and structure from feature correspondences: A review. *Proceedings of the IEEE* **82**(2), 252–268.

Lebégue, X. and J.K. Aggarwal (1993). Significant line segments for an indoor mobile robot. *IEEE Transactions on Robotics and Automation* **9**(6), 801–815.

Luong, Q.T. and O. Faugeras (1996). The fundamental matrix: Theory, algorithms, and stability analysis. *International Journal of Computer Vision* **17**(1), 43–76.

Micheli, E. De, V. Torre and S. Uras (1993). The accuracy of the computation of optical flow and of the recovery of motion parameters. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **15**(5), 434–447.

Mundy, J.L.S. and A. Zisserman (1992). *Geometric Invariance in Computer Vision.* MIT Press. Boston.

Sagüés, C. and J.J. Guerrero (1995). Motion and structure from significant segments in man made environments. In: *IFAC, Intelligent Autonomous Vehicles.* Espoo, Finland. pp. 337–342.