Robotica (2006) volume 24, pp. 113–123. © 2005 Cambridge University Press
doi:10.1017/S0263574704001547 Printed in the United Kingdom

Range image generator including robot motion

Mayra Garduño Gaffare*, Bertrand Vachon† and Armando Segovia de los Ríos‡

(Received in Final Form: December 2, 2004)

SUMMARY
The system here described has the capability of generating range images that include robot motion. The system has two main modules, the motion and the image generator. Motion is modeled using a Bezier's curve method. To compute a range value corresponding to a pixel image, the robot position in the coordinated system is obtained from trajectory generation. In this way, distortion is produced in the image, or sequence of images, as a consequence of motion. The obtained range images represent scenes perceived by the robot from a specific location or during a specified displacement in a very "real" view.

KEYWORDS: Laser range finder; Motion simulation; Range image; Bezier curve.

I. INTRODUCTION

Given the importance of artificial vision in the robotics area and the limitations of the traditional perception systems used in navigation, such as video cameras and ultrasonic sensors, more robust, faster and with better performance systems have emerged to facilitate navigation under difficult conditions. One of these systems has been the laser range finder. These sensors provide 3D information directly from a scene of the real world, however. Though we do not face problems of lower cost navigation systems (video cameras and ultrasonic sensors), with laser telemetry we do face an economic problem. Buying a laser range finder means spending several thousands dollars. Higher costs and little budgets make it difficult to acquire the new and most modern equipment. The above could be the reason of the lack of competitiveness in many laboratories developing advanced techniques. Maybe this is the reason why at this time range images and their processing techniques are hardly mentioned in the literature, including actual computer vision books.[1,2]

This restriction has aroused the interest for developing different low cost tools that allow working with distance data. In the past years, some researches have aimed to reduce the cost of range finder construction for robot's navigation.[3,4]

* Instituto Tecnológico de Toluca, Av. Tecnológico s/n, Metepec, Edo (México).
† Université de la Rochelle, L3i – Pôle Sciences et Technologie, Avenue Marillac, 17042 La Rochelle Cedex 1 (France).
‡ Instituto Nacional de Investigaciones Nucleares, Km. 36.5 Carr. México-Toluca, Salazar, Edo (México).
E-mail: asegovia@nuclear.inin.mx; Corresponding author.

In this sense, the system here presented turns out to be a practical and accessible tool. It is meant to serve as a support for all those investigations that, require images capable of keeping three-dimensional information, i.e. 3D images, known as depth images.

This paper describes the modeling, implementation and experimentation of GIPEM (Generador de Imágenes de Profundidad En Movimiento), a range image generator that includes robot motion. This system is able to generate an image or depth images sequences, linked or not to a robot trajectory. The range data generation depends on the sensor's characteristics, as well as movement specifications and scene description.

There are different methods and technologies for distance data acquisition.[5] This system generates an image, based on the principle of a point laser-based range finder.[6] That is, a single beam is projected and oriented to travel in a horizontal angle Θ and with an elevation angle Φ through $\Delta\theta$ and $\Delta\phi$ increments.

Two important characteristics make GIPEM a very useful experimentation tool. One is the possibility of using real system parameters as ERIM, ODETICS, ALIS, PERCEPTRON, DORNIER, among others,[5,7] and the newer LMS-Z210 High Speed Scanner, from Riegl USA.[4] In this way, it is possible to make a comparative study of applied methods to different range image configurations.

Another important feature of the system is that data are generated from movement. This characteristic is quite important, since it allows one to obtain not only a single image, but also image sequences related to a specified trajectory, such as a real vision system embedded in a vehicle. Resulting images will be affected (deformation of objects) by acquisition times a very important factor with respect to the speed of the vehicle's displacement. This procedure will allow users to study displacement effects in the range image acquisition as it occurs in a real system.

II. FUNCTIONAL ARCHITECTURE OF THE SYSTEM

II.1. Main characteristics of the imaging laser radars

There are three basically different approaches to measuring range:

– Sensor based on measuring the time of flight (TOF) of a pulse emitted energy traveling to a reflecting object and then echoing back to a receiver,

https://doi.org/10.1017/S0263574704001547 Published online by Cambridge University Press

– Phase-shift measurement (or phase detection) ranging technique involves continuous wave transmission, as opposed to the short pulsed outputs used in TOF systems,
– Sensor based in frequency-modulated radar.

Laser-based time-of-flight ranging systems (also known as laser radar or lidar) first appeared in work performed at the Jet Propulsion Laboratory in the 1970's.[8] The time required for a given pulse to reflect off the object and return is measured and used to calculate distance to the target based on the speed of the light. Systems based in this technique, as the SEO LRF-200 OEM (of Schwarts Electro-Optics, Inc.), are capable of giving an accuracy of ±30 cm at ranges up to 100 m.[5]

The phase-shift measurement ranging technique involves continuous wave transmission of an amplitude modulated laser beam directed toward the target. A small portion of this wave coming back to the lidar through Lambertian reflection is used to detect the shifting phase between the emitted beam and its reflection.

Taking into account its angular resolution, one may say that laser telemeters have many physical and technical advantages over other sensors. For example, very fine laser beams may be obtained using small lenses, given that the emitted wave is a short length wave. Besides, coherence spatial quality of the laser sources causes emit beams with constrained diffraction.[9] As an example, an angular resolution of 0.13 mrad may be obtained with a 10 mm diameter beam in a 1 μm wavelength; this means a laser footprint of only 13 mm at 100 m.

With respect to the optical response of the targets, diffusion properties of the materials are favorable to short wavelengths. So, given that the granularity of natural surfaces is superior to laser wavelength, a wide diffusion diagram may be obtained, resulting in a higher detection probability of the laser beam; therefore, it permits one to obtain a laser beam echo at almost any angle.

Another advantage of laser imaging systems is that they are not environmental light dependent; this is the reason why they are more and more on board of mobile robots.[9]

The performance of a 3D-system is tied to four main parameters: range, frame size, precision and frames rate. The range for common systems, as the ALIS,[10] is 1–20 m with a resolution of 20 cm, while for the RIEGL LRS90-3, it is 2–80 m with accuracy of 3 cm.

II.2. Some problems

In general, imaging range finders are very reliable devices, and they are the best sensors to obtain 3-D information. Nevertheless, some inherent problems can significantly impact the quality of the data and limit the use of these sensors. Specifically, four factors are especially interesting: mixed pixels, scanning pattern, range/intensity crosstalk and range drift.[11]

The fingerprint of the projected beam causes the first problem. In fact, mixed pixels are produced by the divergence angle of the laser beam. It is known that a laser pointer is projected in the form of a solid cone, producing a "footprint".[12] So, a beam may detect multiple objects from different distances. For example, if the beam is projected just on the overlap point of two objects that are at very different distances, part of the footprint intercepts an object and the other part will fall on the second object. Then, the value of the measurement will be an average on two very disparate values.[11] This problem originates not perfectly defined distance values, causing disturbances in the real image.

With respect to the scanning pattern one can say that with currently available technology, imaging can be achieved only by scanning the beam using rotating and nodding mirrors. The scanning mechanism introduces additional errors in the sensor. These types of errors are probably the hardest to quantify and to correct. They result in a correct range measurement being stored at the wrong pixel in the image.

Ideally, one would like the range measurements to be completely independent of the reflective properties of the observed objects, but unfortunately, they do influence the range measurements, and in some cases, drastically. The crosstalk effect between range and intensity is due to several causes, but roughly speaking, the lower intensity of the beam coming back to the receiver, the higher the range noise. This affects overall the variance of the measurements, not its mean value. Another source of crosstalk is in the implementation of the detection electronics. Typically, the receiver electronics operate optimally only in a narrow range of intensities compared to the large dynamic range of intensities that can be observed. As a result, surfaces that reflect intensities outside of the optimal operating range will produce noisy or even erroneous range measurements. When the received power is small, the output signal is small, and noise is significant. There are many sources of noise, including photon noise, laser noise, ambient noise, dark-current noise, secondary emission noise, and subsequent amplifier noise. Some experiments have shown that photon noise is the dominant noise source, and that it may be modeled as a Poisson process.

Experiments carried out with Erim and Perceptron imaging laser range finders have shown that significant drift of range measurements appear over working time.[11] As a conclusion of experimentations, it may be said that temperature changes produce the distribution of range values to translate by significant amounts. It is clear that the drift observed must be due to poor temperature compensation in the electronics used in the scanner.

On the other hand, phase based laser range cameras, such as Odetics, ERIM and Perceptron, have problems due to *ambiguity interval*. Conrad and Sampson[13] define this value as the maximum range that allows the phase difference to go through one complete cycle of 360°. This situation introduces a problem for measurements where the round-trip distance exceeds the modulation wavelength. Some applications simply avoid such problems by arranging the optical path so that the maximum possible range is within the ambiguity interval.[5]

In conclusion, one can say that accuracy of the scanner depends in a way on variables such as surface material, ambient illumination and temperature, and that the effect of those variables is amplified by the particular hardware used. In the case of range precision, it has been seen that it decreases with intensity of ambient illumination.

https://doi.org/10.1017/S0263574704001547 Published online by Cambridge University Press

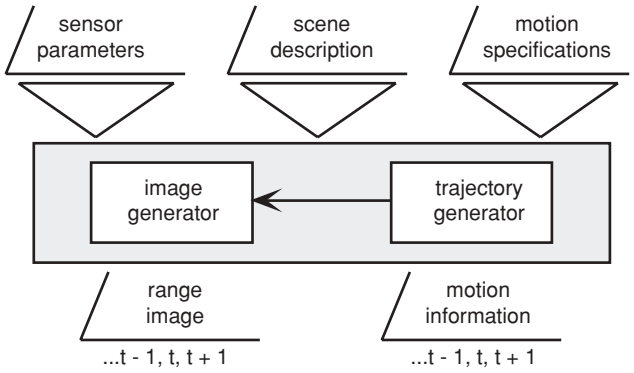


Fig. 1. Range image generation system.

Nevertheless, in despite of all the drawbacks mentioned early, in terms of speed and reliability for medium-range operations, imaging laser range finders are the sensors with the best performance, limiting only its application in many cases the high cost.

II.3. GIPEM organization

It is considered that a mobile vehicle carries the simulated vision system; therefore, images can be acquired in two manners:

- Static Mode (SM). – In this mode, all points forming an image frame are scanned from a fixed position of the robot, and it is then assumed that the robot does not move during image acquisition.
- Dynamic Mode (DM). – In this mode, the robot's position is changing. The dynamical mode can be continuous or static depending on the fact that the scanning image is linked or not to the robot's movement. In the Continuous Dynamical Mode (CDM), the vehicle is displaced during image acquisition and each pixel measured corresponds to a different robot's position. In the Static Dynamical Mode (SDM), the robot executes a trajectory, but it will be stopped at a given distance to scan an image in static mode.

As the Figure 1 diagram shows, there are two basic processes in the system: the image generator and the trajectory generator. These two processes are described in detail in the next two sections. Interaction between these two processes allows the simulator to generate range images that include the vehicle motion. Inputs to the system are sensor parameters, specifications of the movement and description of the scene, described next.

II.4. Parameters required by the system

II.4.1. The scene. The simulator generates images from a previously described artificial scene. The elements constituting it are volumes such as spheres, cylinders and parallelepipeds. These geometric bodies can be combined to represent indoor and outdoor environments. For instance, the combination of several parallelepipeds with different directions and sizes can give the appearance of a house or a box; or well, by combining small cylinders and parallelepipeds, we will be able to create a table or a chair, in addition to many other objects.

The user must describe the scene in a text file required by the system. The file will contain several characters (S, C and/or P) indicating volume and several geometric values (Fig. 2).

In the case of a sphere, a text line containing [S, (x, y, z), r] will describe it. For a cylinder, this line will be [C, (x, y, z), β, α, r, b, h]. Parallelepipeds are described by [P, (x, y, z), h, b, e, β, α].

II.4.2. Motion requirements. Motion is described to the system by the following parameters:

a) First, the robot's displacement must be specified. It is assumed that the robot moves on a plane. Then, its displacement has three degrees of freedom with respect to a fixed universe frame and their descriptors are (x, y) position coordinates and the θ horizontal angle; this conforms the robot's configuration. So, a trajectory is defined by specifying an initial robot configuration (x_i, y_i, θ_i), from which the robot starts its movement and an ending

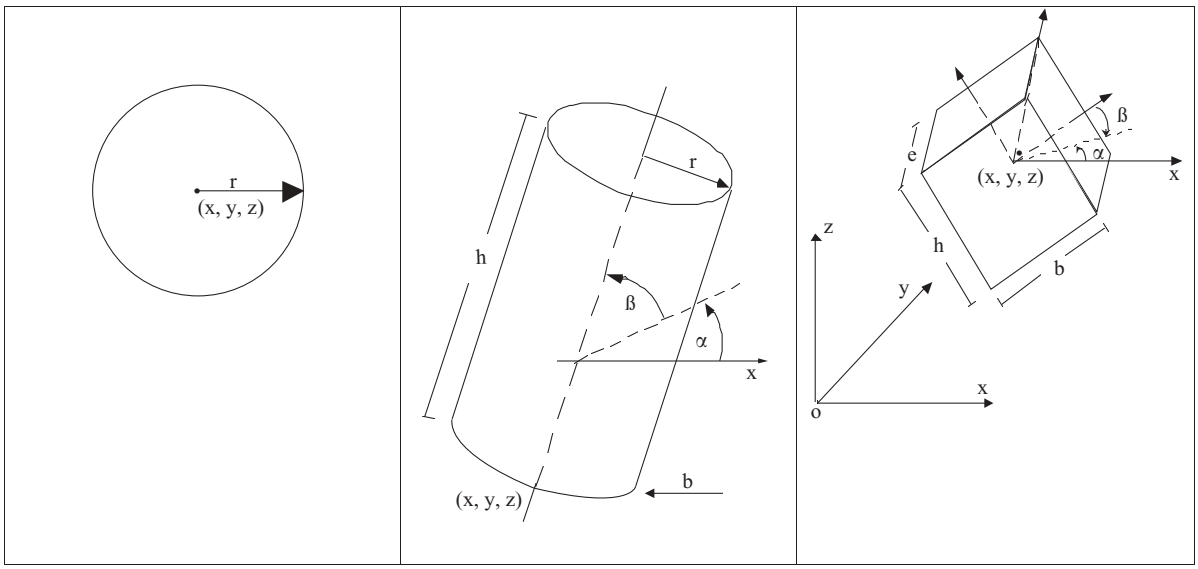


Fig. 2. Volumes geometric description.

https://doi.org/10.1017/S0263574704001547 Published online by Cambridge University Press

configuration (x_d, y_d, θ_d) to which the robot is supposed to arrive.

b) The z-coordinate also required specifies the height of the simulated sensor above the plane. Since the robot is displaced on a plane, variables z_i and z_d are considered constants. Both elevation angles $\phi_i = \phi_d$, are also constant for the system.

c) Another motion parameter is the robot's speed. The user must indicate to system the expected speed to displace the vehicle. This parameter is important, since distortion of the range data varies in function of it. Speed is specified in kilometers/hour.

II.4.3. Sensor parameters. A range-imaging sensor can be characterized by several factors that allows one to measure the sensor performances. The characteristics considered by the simulator in the image generation are the following:

- The Depth of Field indicates the interval over which a scanner measures a distance; it is limited by the minimal and maximal scope of the sensor.
- The Field of View defines the extension of the sensor view limited in two directions and given by the vertical scan angle and the horizontal scan angle.
- The Frame Size gives the number of pixels forming one image, frequently the number of rows and columns that compose the range measurement matrix.
- The Measurement Frequency or Measurement Time is another important parameter that indicates the time in which a measurement can be accomplished. It specifies how many pixels per second a range sensor is capable of scanning. Frequency depends on the efficiency of each particular sensor.

III. MOTION MODEL

The trajectory generator used to introduce motion to the system is based on a Bezier's curve method proposed by one of the authors in earlier works.[14] It is shown that these types of curves are very useful for guiding vehicles because of their important properties.

For path generation, it is required that orientation and curvature be continuous between two arcs. One of the most important advantages is that a complete trajectory can be composed of a sequence of Bezier's arcs, because orientation and continuous curvature are easily obtained by determining their control points adequately.[15] Another important property of Bezier's curves is that the curve is always contained in its polygon of definition. It means that if control points are chosen in such a way that the polygon stays in the free space of the environment, the path will be collision-free.

The effectiveness Bezier's curves for path generation is even shown in more recent works.[16,17]

A Bezier polynomial is a weighted function of a set of control points. This set of points $\mathbf{P}_i$ form a characteristic polygon. The curve is parameterized by u[0, 1] under a polynomial form of order m:

$$\mathbf{P}(\mathrm{u}) = \sum_{i=0}^{m} \mathbf{P}_i B_{i,m}(u) \tag{1}$$

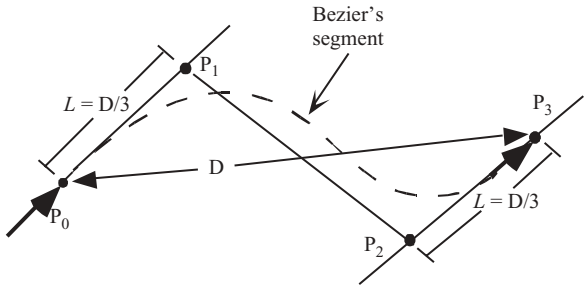


Fig. 3. Bezier's segment definition.

where each polynomial function $B_{i,m}(u)$, named Bernstein's function, is defined as:

$$B_{i,m}(u) = \frac{m!}{i!(m-i)!} u^i (1-u)^{m-i} \tag{2}$$

The minimal solution for initial and ending position and orientation gives a third degree polynomial:

$$\mathbf{P}(\mathbf{u}) = \mathbf{P_0} + 3\mathrm{u}(\mathbf{P_1} - \mathbf{P_0}) + 3\mathrm{u}^2(\mathbf{P_0} - 2\mathbf{P_1} + \mathbf{P_2}) + \mathrm{u}^3(-\mathbf{P_0} + 3\mathbf{P_1} - 3\mathbf{P_2} + \mathbf{P_3}) \tag{3}$$

Then, the arc will be defined by four control points $\mathbf{P_0}$, $\mathbf{P_1}$, $\mathbf{P_2}$ and $\mathbf{P_3}$. Each point is defined by its x, y and z world coordinates. Points $\mathbf{P_0}$ and $\mathbf{P_3}$ are associated to the initial and final robot configuration, respectively, given to the system as motion parameters mentioned on section 2. In this work, $\mathbf{P_1}$ and $\mathbf{P_2}$ control points are computed and distances $\mathbf{P_0P_1}$ and $\mathbf{P_2P_3}$ are fixed to L (see Fig. 3).

The curve segment is obtained evaluating Bezier's polynomial in u. In an image of N rows and M columns, the polynomial is evaluated $N \times M$ times. So, variation of the independent parameter u results in $\Delta u = 1/(N^*M)$. This part will be detailed in section five.

IV. VISION SYSTEM MODELING

IV.1. Sensor geometry description

The geometry of the sensor is presented in Figure 4. The vision system is composed of an optical center Oc and a rectangular image plane P. This image plane can be seen as a grid of L rows by C columns. Each cell in the grid corresponds to a pixel of the image.

It is admitted, without loosing generality, that projection Oc' of Oc on P is the geometric center of P. The sensor coordinated system Rc is determined by its center Oc and by the optical axis $OcOc'$ defining the Xc axis. Yc and Zc axis are selected parallel to the edges of the image plane in such a way that the reference system $Rc = (Oc, Xc, Yc, Zc)$ is direct. The length of vector $OcOc'$, is the focal length df of the sensor.

The field of view is given by angles $\Delta\Theta$ and $\Delta\Phi$, which define the variation limits of orientation angles for the laser beam which measurement is performed, ϕ (elevation angle) and θ (azimuth angle).

https://doi.org/10.1017/S0263574704001547 Published online by Cambridge University Press

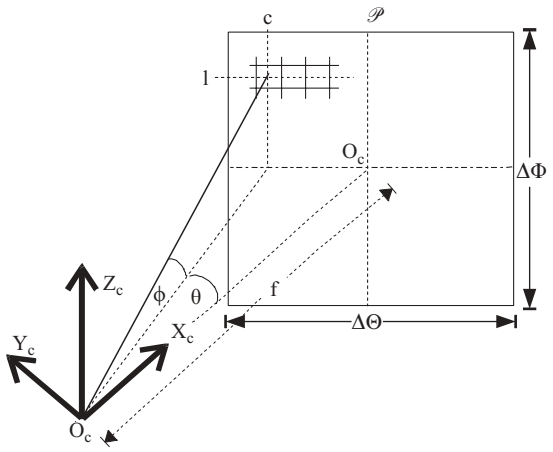


Fig. 4. Vision system geometry.

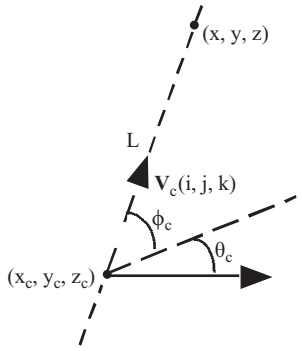


Fig. 5. The laser beam model.

IV.2. The laser beam model of the simulator

A single ray of a point type sensor can be modeled as a line segment that passes through the optical center of the sensor to a cell of the image plane. Knowing the optical center coordinates and the ray orientation, the laser beam is modeled by a line L of parametric equation (see Fig. 5):

$$\begin{aligned} x &= x_c + i\lambda \\ y &= y_c + j\lambda \\ z &= z_c + k\lambda \end{aligned} \tag{4}$$

where:

x_c, y_c, z_c are the robot's position (the robot and sensor coordinates system is considered to be the same).

i, j, k are the unitary components of vector director $\mathbf{V}_c$ of the line L that models the laser beam obtained from the following relations, where (θ_c, ϕ_c) are the direction angles

$$\begin{aligned} i &= \cos\theta_c \cos\phi_c \\ j &= \text{sen}\theta_c \cos\phi_c \\ k &= \text{sen}\phi_c, \end{aligned}$$

considering that: $i + j + k = 1$.

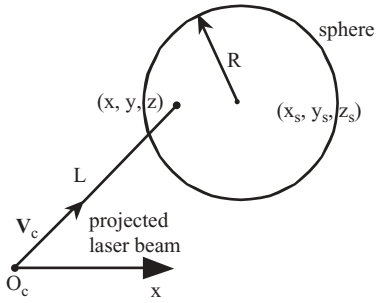


Fig. 6. Sphere description and intersection.

x, y, z are the coordinates of any point belonging to the ray L and,

λ is the scale factor in range, that is, λ gives the distance between the points (x_c, y_c, z_c) and (x, y, z).

IV.3. Spherical object detection

A sphere is detected if the laser beam described by the line segment L has an intersection with a point belonging to such volume. We then specify the parametric sphere equation, which allows determining if there is an intersection with the segment line, as shown in Figure 6:

$$(x - x_s)^2 + (y - y_s)^2 + (z - z_s)^2 = R^2 \tag{5}$$

where:

x_s, y_s, z_s are the coordinates of the center of the sphere,

x, y, z are the coordinates of any point belonging to the sphere surface, and

R corresponds to the radius.

Considering the sphere equation (5) and knowing that the coordinates of a point from the segment line are given by $x_c + i\lambda$, $y_c + j\lambda$, and $z_c + k\lambda$ (equation 4), we can affirm that the intersection of the sphere and the laser beam L is determined by the equation:

$$(x_c + i\lambda - x_s)^2 + (y_c + j\lambda - y_s)^2 + (z_c + k\lambda - z_s)^2 - R^2 = 0$$

where we obtain a quadratic equation of the form $A\lambda^2 + B\lambda + C = 0$:

$$\begin{aligned} &\lambda^2 + 2\lambda[i(x_c - x_s) + j(y_c - y_s) + k(z_c - z_s)] + (x_c - x_s)^2 \\ &\quad + (y_c - y_s)^2 + (z_c - z_s)^2 - R^2 = 0 \end{aligned} \tag{6}$$

In order to determine whether or not there exists an intersection, we have analyzed the nature of the roots obtained with the following assumptions (see Fig. 7):

Case 1: not real roots, this means that the beam has no intersection with the sphere.

Case 2: two equals real root, the beam is a tangent line to the sphere (point T).

Case 3: two unequal real roots, which mean that the beam crosses the sphere at points P and P′.

The first case, where no solution is found, means that the laser beam at the corresponding (i, j, k) orientation does not visualize any spherical object. The pixel will then receive

https://doi.org/10.1017/S0263574704001547 Published online by Cambridge University Press

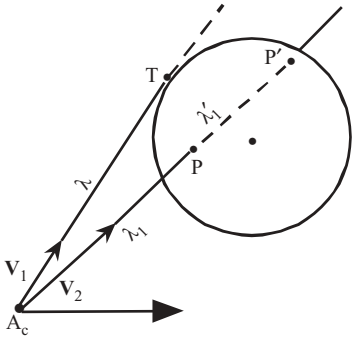


Fig. 7. Intersected points.

the equivalent infinite value, that is, it is assigned the darkest gray level.

When the result obtained matches a double root as in the second case, the ray intersects a point of the periphery of the sphere. The detected point then takes the only value found of λ, $\lambda = A_cT$.

In the last case, λ gives two distances $A_cP = \lambda_1$ and $A_cP' = \lambda'_1$, as shown in Figure 7. One of these distances is the value associated to the pixel of the image. The sign of the solution (or solutions) is important and indicates that:

- Negative values. If both values are negative, the intersection point found will not belong to the line segment that models the laser beam, but to an impact point corresponding to an object behind the sensor. In this case, the darkest gray level is assigned to the pixel.
- Negative and positive values. If the result has negative and positive values, it means that the sensor is found within the sphere. This particular case is invalid (for outdoor navigation), since the vehicle has penetrated the object. As in the previous case, the darkness gray level is assigned to the pixel.
- Positive values. Finally, if two positive values are obtained, the sphere is seen from the front, and two intersected points, P and P′, are found. The impact point P′ belongs to the hidden face of the object, and that is the reason why the largest distance is discarded. The smallest distance, corresponding to the point impact P, is the real true value. In this case, the pixel gray level value is given by λ_1.

The coordinates (x, y, z) of the impact point can be determined with equation (4), substituting the appropriate value of λ.

IV.4. Detection of a cylindrical object

The general equation that defines a cylinder in the space 3-D is given as:

$$(x - x_{cil})^2 + (y - y_{cil})^2 + (z - z_{cil})^2 - [\alpha(x - x_{cil}) + \beta(y - y_{cil}) + \gamma(z - z_{cil})]^2 = R^2 \quad (7)$$

where:

$x_{cil}, y_{cil}, z_{cil}$ are the coordinates of a point belonging to the central axis of the cylinder.

α, β, γ are the unitary components of the axis of the cylinder

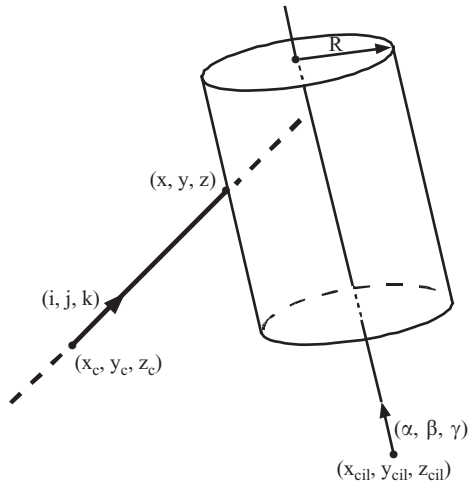


Fig. 8. Cylinder and beam intersection.

x, y, z are the coordinates of any point that belongs to the cylinder, and

R corresponds to the radio of the volume (see Fig. 8).

From equations 4 and 7, we deduce that the intersection between a cylinder and a laser beam L is determined by the equation:

$$(x_c + i\lambda - x_{cil})^2 + (y_c + j\lambda - y_{cil})^2 + (z_c + k\lambda - z_{cil})^2 - [\alpha(x_c + i\lambda - x_{cil}) + \beta(y_c + j\lambda - y_{cil}) + \gamma(z_c + k\lambda - z_{cil})]^2 = R^2 \quad (8)$$

from which a second degree equation is obtained, as for the sphere case:

$$\begin{aligned}&\lambda^2\{i^2(1-\alpha^2) + j^2(1-\beta^2) + k^2(1-\gamma^2) - 2\alpha\beta ij - 2\alpha\gamma ik \\ &- 2\beta\gamma jk\} + \lambda 2\{i(x_c - x_{cil})(1-\alpha^2) + j(y_c - y_{cil})(1-\beta^2) \\ &+ k(z_c - z_{cil})(1-\gamma^2) - \alpha\beta[j(x_c - x_{cil}) + i(y_c - y_{cil})] \\ &- \alpha\gamma[k(x_c - x_{cil}) + i(z_c - z_{cil})] - \beta\gamma[k(y_c - y_{cil}) \\ &+ j(z_c - Z_{cil})]\} + \{(x_c - x_{cil})^2(1-\alpha^2) \\ &+ (y_c - y_{cil})^2(1-\beta^2) + (z_c - z_{cil})^2(1-\gamma^2) \\ &- 2\alpha\beta(x_c - x_{cil})(y_c - y_{cil}) - 2\alpha\gamma(x_c - x_{cil})(z_c - z_{cil}) \\ &- 2\beta\gamma(y_c - y_{cil})(z_c - z_{cil}) - R^2\} = 0. \end{aligned} \quad (9)$$

This equation may be represented as $\lambda^2 A_c + \lambda B_c + C_c = 0$.

Evaluating A_c, B_c, and C_c, it is possible to determine λ with the same three sphere cases. It is also possible to calculate coordinates (x, y, z) of the intersection point with equation (4).

Now then, if there is a point p of (x, y, z) coordinates such that p belongs to the cylinder and p belongs to the beam D, then it should be determined if p is within the area of the designed cylinder. That is, since equation (7) defines an unlimited cylinder, the intersection point $p(x, y, z)$ found must be evaluated on a limited area forming the cylinder. Two parallel planes represented by the corresponding equations

https://doi.org/10.1017/S0263574704001547 Published online by Cambridge University Press

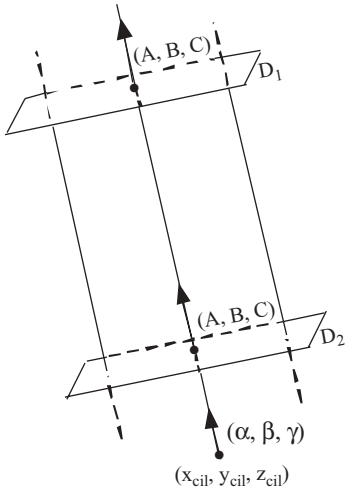


Fig. 9. Cylinder limits.

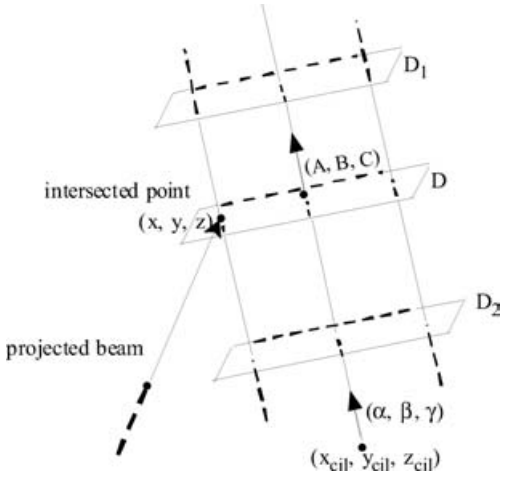


Fig. 10. Determining if an intersected point belongs to the cylinder.

limit the cylinder

$$Ax + By + Cz = D_1$$
$$Ax + By + Cz = D_2$$

Both planes are perpendicular to the cylinder central axis. Each particular plane will be then obtained fixing the value D_1 and D_2, where D_1 corresponds to the top of the cylinder and D_2 to the bottom (Fig. 9). These two parameters, D_1 and D_2, are the value of height and base, respectively, given in the scene description file.

When the cylinder is limited, we then verify if the intersection point is within the surface limited by D_1 and D_2. For this, we obtain a new equation $D = Ax + By + Cz$ from a third imaginary plane to which the intersected point belongs (Fig. 10).

In this figure, *A*, *B* and *C* are the unitary components of the normal vector of the plane and *x*, *y*, and *z* are the coordinates of an intersected point calculated by equation (4). As we can see in Figure 10, plane D is perpendicular to the cylinder central axis that defines the central plane of the cylinder; therefore, (A, B, C) = (α, β, γ).

Once D is obtained, it is determined that if:

$$D_2 \leq D \leq D_1$$

Then the intersection point x, y, z belongs to the cylinder and it is then coded according to the distance determined by λ.

IV.5. Parallelepiped detection

Three pairs of parallel planes can define a parallelepiped. As seen in the previous section, a plane can be represented by an equation with the form:

$$Ax_p + By_p + Cz_p = D \tag{10}$$

A plane family can be defined by their common vector normal. A particular plane is obtained fixing the value of D, knowing the coordinates of a point belonging to this plane.

The three pairs of planes used to define a parallelepiped are perpendicular between them. A parallelepiped is defined by its center P_p of (x_p, y_p, z_p) coordinates, by a normal vector to a pair of planes oriented by α y β angles, and by three values *h*, *b*, and *e* that determine height, length and width of the volume, respectively (Fig. 11).

From here, it is possible to calculate the three normal vectors that define the direction of each pair of planes, thus having $\mathbf{N}_1$, $\mathbf{N}_2$ y $\mathbf{N}_3$ as the corresponding normal vectors of parallelepiped faces 1, 2 and 3, respectively. The normal vectors of the remaining faces (4, 5, 6) are defined according to the vector of its respective parallel face, since: $\mathbf{N}_i = -\mathbf{N}_{i+3}$, where ($1 \leq i \leq 3$), (Fig. 12).

To determine the planes that define the parallelepiped faces, an imaginary central reference plane D to which point P_p belongs is determined. Suppose that we want to

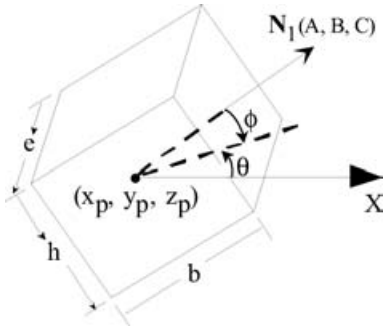


Fig. 11. Descriptive parameters of a parallelepiped.

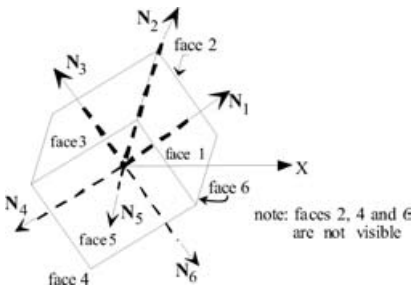


Fig. 12. Parallelepiped model.

https://doi.org/10.1017/S0263574704001547 Published online by Cambridge University Press

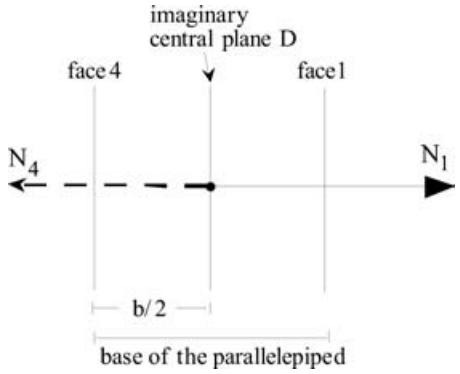


Fig. 13. Determination of parameters D for each pair of plans.

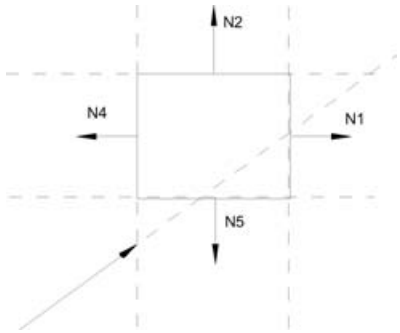


Fig. 14. Visibility of the volume faces.

find faces 1 and 4 of the normal vector $\mathbf{N}_1$ that determine the parallelepiped base *b*. Evaluating equation (10) with $(A, B, C) = \mathbf{N}_1$, we obtain the central reference plane D (Fig. 13).

From D, we determine values D_1 and D_2 that correspond to face 1 and 4, respectively, being:

$$D_1 = D + b/2, \qquad D_4 = -(D + b/2).$$

When the parallelepiped is defined, we determine the moment when it is detected by the laser beam. This is determined by calculating λ with the following equation, obtained from equations (4) and (10):

$$\lambda = \frac{D - Ax_c - By_c - Cz_c}{Ai + Bj + Ck} \tag{11}$$

This equation must give a unique λ value to be assigned to the evaluated pixel, however, determining a unique value results less evident than for spheres and cylinders. In a real case, there is only one visible surface to the vision system, but this is not true for the simulator. In the model of the scene, the planes constituting the parallelepiped are infinite surfaces, and the beam will intersect at the same time at least two planes. For example, when analyzing the case presented in Figure 14, we observe that the beam has intersection points with planes defined by vectors $\mathbf{N}_4$, $\mathbf{N}_5$, $\mathbf{N}_1$, and $\mathbf{N}_2$. However, the only visible surface for the sensor is that defined by vector $\mathbf{N}_5$, and the λ true value is the distance measured between the origin beam and the intersected point at surface 5.

Analyzing the model of Figure 14, we see that the scalar product will indicate if the detected plane is a hidden or visible face to the sensor. With this purpose, the scalar product between the line segment vector modeling the laser beam and the normal $\mathbf{N}_i$ is calculated. A scalar product result negative means parallelepiped's face is really "seen" by the sensor.

On the other hand, as already said, equation 10 defines infinite flat surfaces, however, only part of these surfaces defines the parallelepiped. Once determined that the intersected point belongs to visible surfaces, it is necessary to determine if the point detected is within the parallelepiped boundaries. As seen on Figure 14, the intersection point at the plane defined by vector $\mathbf{N}_4$ is found outside the volume dimensions. Furthermore, the intersection with plane $\mathbf{N}_5$ is found within the space defined by planes $\mathbf{N}_1$ y $\mathbf{N}_4$. The solution is as outlined to determine if an intersected point is or not within the boundaries of the cylinder (see section IV.4).

V. RANGE IMAGE GENERATION

V.1. Rendering the image

When the vision system model and the path generator method are defined, one image is computed through the following process.

A range value is measured when a robot's configuration is determined in a specific location of the trajectory. The sensor system coordinate R_c is referenced in the world coordinates system R_w, and the imaginary image plane is fixed on the space.

To measure the range value corresponding to pixel *p*, a position of the robot in the world coordinates system R_w is obtained from trajectory generation. The sensor and the robot are considered to be linked, and geometrical transformation from R_c to R_w is required. The focal distance $\mathbf{df} = \mathbf{O_c O'_c}$ is calculated by trigonometric relations according to the parameters of the selected laser system.

From the robot's configuration, a pixel *p* of (*r*, *c*) coordinates is fixed on the image plane (figure 15a). The vector director at pixel *p* determines the orientation of the beam supporting the measure. A projected beam has its

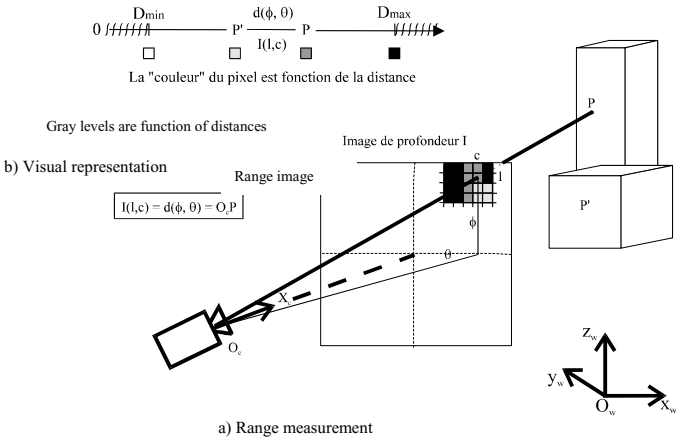


Fig. 15. World representation on the image plane.

https://doi.org/10.1017/S0263574704001547 Published online by Cambridge University Press

origin at the optical center *Oc* and passes through the pixel center.

Before measuring the range value, the geometrical transformation from R_c to R_w is carried out in order to reference all measurement to a universal coordinated system.

Knowing the ray direction, it is evaluated with each object in the scene to determine if there is an intersection with some of them or not; if there is, the distance $\mathbf{O_cP}$ will then give the value of the corresponding pixel. As with physical sensors, the simulator completes one image scanning the whole view field from left to right and from top to bottom. The increment values are determined according to the number of rows and columns on the image plane.

When one image is acquired in Continuous Dynamical Mode (CDM) (section II.1), measurements are pixel by pixel, taking into account time, speed and positioning factors. That is, according to the laser frequency selected and to displacement time employed by the robot to execute a trajectory it is possible to determine the number of pixels that the system can compute from the specific path, considering that:

$$\text{Total_Pixels} = \text{Laser_Frequency} * \text{Displacement_Time}$$

where the Displacement_Time will be a function of the traveled distance and the robot speed. Determining the total pixels, the evaluation Bezier's curve parameter **u** is easily obtained. Thus, knowing that $u \in [0, 1]$, the Δu increments are $\Delta u = 1.0/\text{Total_Pixels}$.

The vision system starts acquisition of an image at initial robot's configuration and computes the pixel $d(0, 0)$. A Bezier polynomial is evaluated to give the next configuration of the robot and to compute pixel $d(0, 1)$. The process is repeated for each new robot's configuration, at each new position, and a new range is calculated so that each pixel will have a value measured from a particular position and orientation of the robot configuration. The process will be stopped when the vehicle finishes its trajectory, and the number of generated images will depend on the displacement robot's speed and on the laser frequency of the selected sensor. If, at the end of the trajectory, one image is not completely calculated, the remaining points will be measured in static mode from the final position of the robot.

V.2. Considering noise

The development and perfection of the laser have allowed considerable improvements in the precision and sensibility of optical measurements. However, validity of the results is limited by noise, as explained in section II.2. The most concerning problems discussed in researches related to laser telemetry are those of mixed pixels and ambiguity,[11] caused by the beam footprint and the wavelength defined by the modulation of the laser beam, though other factors are also important. Some of these have been characterized in a more or less precise way by a distribution of probability, as is the case for the photon noise of the receiver, where a Poisson process has been assumed.[18] The rough data of a real image are affected to a large extent all over these factors. Considering that the simulation in this work seeks to be as real as possible, a module of the simulator allows introducing a parametric noise to the generated images, trying to take into account all the noise sources of a real range imaging system. So, a Gaussian noise r of zero mean is associated with each measurement d(l, c) and with each angle θ and ϕ. The standard deviation of r is variable and depends on the precision of the laser. The resulting distance d_r will be $d(1 + r_0)$ given by a direction $\theta(1 + r_1)$ and $\phi\ (1 + r_2)$.

Finally, distances so obtained are codified between 0 and 255 gray level values. Darker colors correspond to the farther objects and brighter colors correspond to the nearest objects from the sensor. If no intersection can be found, the maximum range value is assigned to the pixel.

VI. RESULTS

Originally, the system was developed in *C++* to work in a Sun workstation, and after it was upgraded to work in a Silicon Graphics station, always running in a like UNIX Operating System. Recently, in order to obtain transportability to other platforms and a simpler program, the system has been translated to *Java*. The graphical user interface is shown in Figure 16. This is the main window where the user gives the scene and motion to the system and selects a specific sensor.

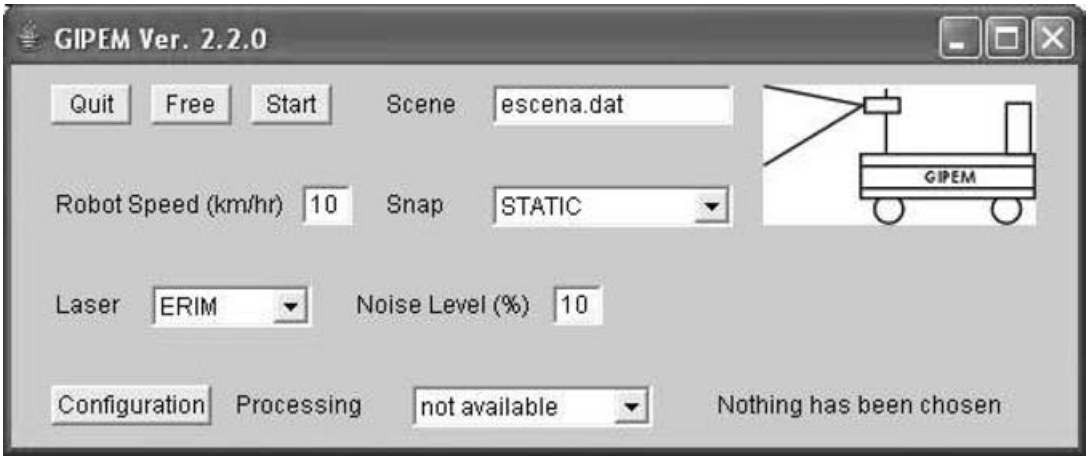


Fig. 16. Main window of the system.

https://doi.org/10.1017/S0263574704001547 Published online by Cambridge University Press

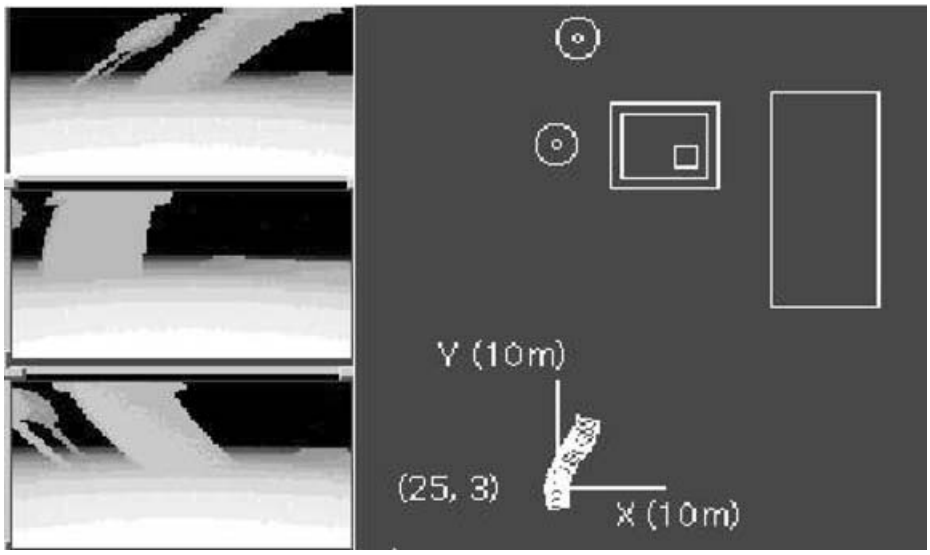


Fig. 17. Range image sequence and the corresponding robot trajectory.



Fig. 18. Example of an image issued of the Ben Franklin Laser Scanner.

The imaging capabilities of the sensor are demonstrated by an example. Figure 17 shows a sequence of three images taken from the trajectory executed using the parameters of the imaging range system DORNIER. The scene is composed of four objects visualized in images of 128×64 pixels, scanned in $70^\circ \times 32^\circ$ of field of view. White corresponds to the smallest distance and black to the greatest, as in the case of the Ben Franklin Laser Scanner,[19] which has a similar ERIM performance and a programmable scan and is a more modern laser range finder. An image issued of this real laser range finder is shown in Figure 18, to compare appearance with Figure 17, issued by the simulator. As one can see, results are very real, and the image in figure 17 lets one to observe deformation caused by the displacement of the "robot".

VII. CONCLUDING REMARKS

This paper describes a range image generator. This simulator is able to model a three-dimensional sensor carried on a mobile vehicle. It is based on a point laser principle, which means that it measures the distance along a single vector director from an assumed optical center to a visible surface point in the scene.

A range image represents a view by the sensor of a synthetic scene composed of several volumes. The data we obtain is a function of sensor and motion specifications. Thus, simulation is based on the characteristics of real sensors and other parameters as robot's speed and displacement. The most important characteristic of the system is that it involves distorted images, due to the time and the motion between scanned pixels. This system actually serves as support for several investigations at our laboratory. It has allowed us test the currently available algorithms for digital processing of range images, and it has permitted to put in manifesto the existing problem on the distorted images produced by real sensors when the vehicle is moving. Otherwise, it is a solution to the still very high cost of modern laser range finders to realize researches in a non very exploited area of range data supplied by range sensors.

Acknowledgement

The authors wish to thank CONACYT and ANUIES (Mexico, project M01-M01) for the support to the work presented here. They wish to thank also COSNET (Mexico) for the financial support in part of this work (project 639.02-03-PR).

References

1. Emanuele Trucco and Alessandro Verri, *Introductory Techniques for 3-D Computer Vision* (Prentice-Hall, 1998).
2. David A. Forsyth and Jean Ponce, *Computer Vision: A Modern Approach* (Prentice Hall, 2002).
3. G. L. Miller and E. R. Wagner, "An Optical Range Finder for Autonomous Robot Cart Navigation," *Autonomous Robot Vehicles* (Springer-Verlag, New York, 1990).
4. www.rieglusa.com.
5. J. Borenstein, H. R. Everett and L. Feng, *Where am I?- Sensors and Methods for Mobile Robot Positioning* (The University of Michigan, April 1996).
6. P. Besl, "Active Optical Range Imaging Sensors," *Machine Vision and Applications* (Springer-Verlag, New York Inc., 1988).
7. J. P. Cariou, "Capteur de perception," In: *Robotique Mobile* (Eds. C. Fargeon and J. P. Quin) (Ed. Teknea, Toulouse, France, 1993).
8. R. A. Lewis and A. R. Johnson, "A Scanning Laser Range Finder for a Robotic Vehicle," *5th International Joint Conference on Artificial Intelligence* (1977) pp. 762–768.
9. C. Fargeon and J. P. Quin, *Robotique Mobile* (Ed. Teknea, Toulouse, France, 1993).

https://doi.org/10.1017/S0263574704001547 Published online by Cambridge University Press

10. P. Besesty, *ALIS, Advanced Laser Imaging System* (CEA-LETI, France, 1991).
11. M. Hebert and E. Krotkov, "3-D Measurements From Imaging Laser Radars: How Good Are They?," *IEEE/RSJ International Workshop on Intelligent Robots and Systems IROS'91*, Osaka, Japan (Nov., 1991) pp. 359–364.
12. P. A. Veatch, and L. S. Davies, *Range Image Algorithms for the Detection of Obstacles by Autonomous Vehicles* (Center for Automation Research, Maryland University, July 1987).
13. D. J. Conrad and R. E. Sampson, "3D Range Imaging Sensors," In: *Traditional and Non-Traditional Robotic Sensors* (edited by Th. C. Henderson) (NATO ASI Series, Springer-Verlag, 1990) Vol. F63, pp. 35–47.
14. A. Segovia, M. Rombaut, A. Preciado and D. Meizel, "Bezier's Polynomials: A solution for a Mobile Robot Path Planning," *Workshop on Mobile Robotics for Civil Works, Brussels* (June 6–7, 1991) pp. 1–11.
15. A. Segovia and M. Rombaut, "Continuous Curvature Path Finding for a Non-Holonomic MobileRobot," *1st International Workshop Intelligent Autonomous Vehicles* (Pergamon Press, 1993) pp. 481–486.
16. A. Arakawa, M. Hiyama. T. Emura and Y. Kagami, "Trajectory Generation for Wheeled Mobile Robot Based on LandMarks," *0-7803-2559-1/95 IEEE* (1995) pp. 1273–1278.
17. N. Le Fort-Piat, I. Collin and D. Meizel, "Planning robust displacement missions by means of robot-tasks and local maps," *Robotics and Autonomous Systems* **20**(1), 99–114 (1997).
18. D. Nitzan, A. Brain and R. Duda, "The Measurement and Use of Registered Reflectance and Range Data in Scene Analysis," *Proc. IEEE* (February, 1977) **Vol.** 65(2) pp. 206–220.
19. www.frc.ri.cmu.edu/∼alonzo/projects/bfscanner/bfscanner.html.

https://doi.org/10.1017/S0263574704001547 Published online by Cambridge University Press