UNIVERSITÉ DE NICE-SOPHIA ANTIPOLIS

Année universitaire 1994-1995

DEA Théorie et Applications
de la Vision Artificielle et de la Robotique

SPECIFICATION OF SENSOR-BASED TASKS WITH A VIEW TO AN APPLICATION TO MOTION PLANNING

Laurence PELLETIER

Sous la direction de Patrick RIVES

Abstract

This study takes place in the general framework of the INRIA project *Praxitèle* which aims at automatically locating and guiding from sensing information an autonomous electric car.
It deals with the relation between sensor-based control and sensor-based motion planning for an a priori known environment, presenting a method to combine robustness in control with planning purposes.

The first part of this work consists in determining a set of sensor-based primitives for range sensors and implementing generic computation of interaction screws within the framework of sensor-based control.
The aim is to extend the works realized in the case of vision sensors for range sensors, in order to get a catalog of sensor-based action primitives for specifying local robot tasks.

The second part attempts to demonstrate that we can use the local constraints arising from the task at hand to simplify global motion planning into sensor-based task planning.

Keywords : Sensor-based control, interaction screw, range sensors, task geometry, sensor-based motion planning.

Contents

Notations

Throughout this dissertation we will note :

$\langle\,.\,|\,.\,\rangle$	the scalar product
$\times$	the vector cross product
$.$	the matrix product
$\bullet$	the screw product
$\|\,.\,\|$	the euclidian norm of a vector
O	the origin of the fixed frame of the 3D scene
S	the source (of the sensor)
$T, T_i, T_{i,j}$	the target
F_R	the reference frame of the 3 D scene (fixed)
F_T	the frame (mobile or fixed) linked to the target
F_S	the mobile frame linked to the sensor
$n_T, n_{T_i}, n_{T_{i,j}}$	the unitary vector normal to the surface of the target
$n_S, n_{S_i}, n_{S_{i,j}}$	the unitary vector characterizing a pin-point sensor
$\delta, \delta_i, \delta_{i,j}$	the distance measured between the pin-point sensor and the target
s	the signal delivered by the sensor
s_d, s_{id}	the desired sensor value
T_{ST} , or	the screw representing the location of the frame F_S with respect to the frame F_T
$(V_S, \omega_{F_S/F_T})$	where V_S and ω_{F_S/F_T} represents its translational and rotational components respectively

Chapter 1

Introduction

Our work takes place within the general framework of the INRIA project *Praxitèle* which aims at locating and automatically guiding from sensors' information an autonomous electric car. Such an automatic guided vehicle (A. G. V.) can be considered as a non holonomic robot moving in a dynamic semi-structured environment, guided thanks to the information provided by on-board sensors.
These are used in order to enable the robot to react with its environment: on-line obstacle avoidance, execution of tasks, updating its knowledge of the world. . .

This is typically a problem of Perception-Based Navigation, a subject very in fashion but highly complex. It can be treated from two standpoints :

- the control standpoint: very efficient for local approaches, dealing with dynamic obstacle avoidance [Nov94] or with robust control schemes for sensor-based control [Cha90, PG93].

- the motion planning standpoint : a global approach based on a geometric modelling of the scene and of the robot [Lat90, Bui94] with the use of geometric models of the robot in the control loop.

From the control point of view that we will adopt later on, this second kind of approaches leads to errors in control due to errors of modelling which must be corrected somehow.

Sensing is used not only to update the knowledge of the environment, to deal with uncertainties or to react in a partially known or even dynamic environment, but also to guide the motions and monitor their execution. To achieve this goal, exteroceptive sensors (which give information about the world) are more and more used since proprioceptive sensors (which give information about the internal state of the robot) are not sufficient in the case of an environment not precisely known a priori and might accumulate errors of localization.
Unfortunately, no single sensor can by itself achieve both accuracy and speed in sensing and processing ([KTB87]), and sensory input data is typically local. For instance, range or proximity sensors provide useful depth information for on-line obstacle avoidance whereas vision sensors enable to locate in an a priori known 3D scene [ZF92]. Often several sensors are needed, not necessary of the same type. The

problem of merging the different sensory data and the prior knowledge of the world, sensor fusion, needs a particular effort of modelling.

As the planning problem is highly complex, some people consider only sensing for on-line dynamic obstacle avoidance [PS95, Fra91, FS94, KL] but without considering the sensor data to update the knowledge of the global environment. Others [BF95, Kha86] take into account sensory data but do not consider non holonomic constraints. Approaches trying to unify both questions of dealing with the constraints of the robot and taking into account sensor information are task planning or sensing planning [PS95, FH93, LS95, MI95].

Indeed, uncertainty often leads to the use of sensor-based motion commands whose behavior is less sensitive to errors than purely position-controlled motion commands. We place our work within the general framework of the sensor-based control defined by [ECR90] using the task function approach [SEB90]. Then, we extend the works previously developed [CRE93, Cha90, PG93] to the case of range sensors and exploit geometrically our results. Finally, we point out how sensor-based control helps to transform a motion planning problem into a sensor-based task planning one for an a priori known static environment.

Chapter 2

Setting the problem

2.1 Assumptions on the 3D scene

In order to move safely every mobile robot needs having a map of the world based on sensor information, a priori knowledge and generic models. Since a parking for instance, is a partially known, semi-structured environment, to simplify our study we will from now on focus on an a priori known structured outdoor environment.

Throughout this report, we will consider an *a priori* known geometric 3D scene, composed of a set of objects described by their surface.

Let us start with our description of the world by setting the equations of the geometric features we will take into account in the scene :

- 3D Points:
  A 3D point is represented by its coordinates (x, y, z), which are are given, unless specified, in a fixed reference frame F_R of the 3D scene.

- 3D Lines:
  A 3D line is defined by the intersection of two planes. In the case of a 3D straight line, the two planes can be chosen arbitrarily, but in the case of an edge the two planes are the incident faces which give rise to this edge.

$$\begin{cases} a_1x + b_1y + c_1z + d_1 = 0 \\ a_2x + b_2y + c_2z + d_2 = 0 \end{cases}$$

- Planes :
  A 3D plane is given by its equation :

$$ax + by + cz + d = 0$$

- Spheres :
  Let us consider a sphere of center (x_0, y_0, z_0) and radius R, its cartesian equation is:

$$(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 - R^2 = 0$$

- Cylinders :
  Let $\mathcal{C}$ be a cylinder of generating axis defined by the direction $(a\ b\ c)^T$ (with $a^2 + b^2 + c^2 = 1$) and the point (x_0, y_0, z_0) belonging to the plane $\mathcal{P}$ orthogonal to this direction passing by the origin.

  So we have :

$$a\, x_0 + b\, y_0 + c\, z_0 = 0$$

$\mathcal{C}$ is the set of all the circles $\mathcal{C}_i$ of center R passing by a point (x_i, y_i, z_i) of the generating axis, and belonging to a plane parallel to $\mathcal{P}$.
So $\mathcal{C}_i$ is the intersection of the sphere of center (x_i, y_i, z_i) and radius R and the plane of normal axis $(a\ b\ c)^T$ passing by (x_i, y_i, z_i), which we can express as :

$$\begin{cases} (x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 - R^2 = 0 \\ a\,(x - x_i) + b\,(y - y_i) + c\,(z - z_i) = 0 \end{cases}$$

Expressing that (x_i, y_i, z_i) belongs to the axis defined by the point (x_0, y_0, z_0) and the direction $(a\ b\ c)^T$ yields :

$$a(x_i - x_0) + b(y_i - y_0) + c(z_i - z_0) = 0$$

Eliminating x_i, y_i and z_i between these equations we finally obtain the equations of $\mathcal{C}$:

$$(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 - [a(x - x_0) + b(y - y_0) + c(z - z_0)]^2 - R^2 = 0$$

with

$$\begin{cases} a^2 + b^2 + c^2 = 1 \\ a\,x_0 + b\,y_0 + c\,z_0 = 0 \end{cases}$$

We chose implicit equations to represent the 3D features in the scene because the computation of their normal vector is then obtained straightforward by computing a gradient. Indeed, to compute the normal vector at a point T of a 3D feature of implicit equation $\mathcal{F}(x, y, z) = 0$ we apply the gradient of the function $\mathcal{F}$ at this point T:

$$\vec{n_T} = \nabla\mathcal{F}(x_T, y_T, z_T)$$

.

2.2 Modelling interactions of the robot with its environment

In order to use a sensor-based closed loop control scheme, we will consider only sensors able to provide a $\mathcal{C}^2$ (*id est* continuous, twice differentiable) feedback vector. This means considering high rate data sensors (because of the dynamics of the system, so as to ensure stability) and sensors not providing symbolic data (because of real-time constraints).

Let us consider a sensor S rigidly fixed to a robot, with associated frame F_S . This sensor perceives a target T in the environment, with associated frame F_T. It provides a vectorial signal s, which can be decomposed into scalar signals s_i . We will assume later on that this signal s_i depends only on the relative location of F_T with respect to F_S. This means that if the sensor is motionless with respect to the target, then the signal remains constant. This assumption is valid in practice for a large class of sensors providing geometric information, such as force sensors, vision sensors or range sensors. For our purposes we will consider range sensor, extending the approach developed by F. Chaumette ([Cha90, CRE93]) for a projective camera.

The relative location (position and orientation) of F_T with respect to F_S can be parameterized by an element $\underline{r}$ of the Lie group of displacements SE_3. $\underline{r}$ depends on the robot joint coordinates $\underline{q}$ and time t. We may thus write:

$$s_i(\underline{r}(\underline{q}, t)) = s_i(F_S, F_T) \tag{2.1}$$

where s_i is a $\mathcal{C}^2$ function defined over SE_3, and valued in $\mathbb{R}$.Then

$$\dot{s}_i = \frac{\partial s_i}{\partial \underline{r}}\,\dot{\underline{r}} \tag{2.2}$$

Moreover, se_3, the Lie Algebra of SE_3 (which is its tangent space of at the identity), can be assimilated to the space of screws. A screw is a vector field V on $\mathbb{R}_3$ for which there exists a vector v and a point O such that for all points M in $\mathbb{R}_3$,

$$V(M) = V(O) + v \times \vec{OM}$$

The differential of s_i is a linear mapping from se_3 to $\mathbb{R}$, thus it belongs to its dual, $(se_3)^*$, isomorphic to se_3. It can be represented by a screw $H_i = (H_i(S), u)$ so that

$$\frac{\partial s_i}{\partial \underline{r}}\,(V, \omega) = (H_{i,j}(S), u) \bullet (V, \omega) \tag{2.3}$$

For $\dot{\underline{r}} = T_{ST}$ we finally obtain:

$$\dot{s}_i = H_i \bullet T_{ST} \tag{2.4}$$

where the product of screws is the scalar $\langle\, H_i(S) \,|\, \omega \,\rangle + \langle\, V(S) \,|\, u \,\rangle$.

We can note that H_i represents uniquely the interaction between the sensor and the robot's environment, so it is called *Interaction Screw*.

For a particular value s_{i_d} of the signal associated to a given location $\underline{r}$, there exits some displacements T^*_{ST} of the sensor with respect to the target for which the value s_{i_d} of the signal remains unchanged. Given the interaction screw H_i, the velocity screws that leaves $s_{i,j}$ invariant are solutions of:

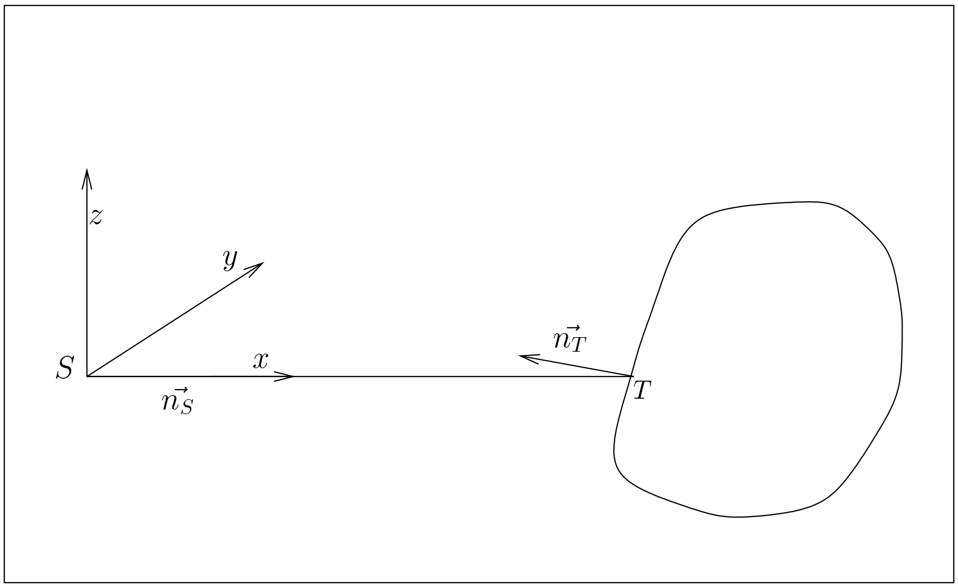
$$H_{i,j} \bullet T^*_{ST} = 0 \tag{2.5}$$

T^*_{ST} is said to be *reciprocical* to H_i.

In the general case, if we denote by $\underline{s}$ the vector regrouping all the signals $s_{i,j}$, the displacements T^*_{ST} of the sensor around the location $\underline{r}$ which leave $\underline{s}$ invariant are characterized by the reciprocical subspace S^* of the subspace spanned by the set of screws (H_i).

2.3 Models of the sensors

2.3.1 Computation of the Interaction Screw for an elementary pin-point sensor



Let us consider the signal

$$s = f(\delta) \tag{2.6}$$

where $\delta = ST$ and f is a linear function.
We look for the Interaction Screw $H = (H(S), u)$ such as $\dot{s} = H \bullet T_{ST}$, where $T_{ST} = (V_S, \omega_{F_S/F_T})$.

Equation (2.6) yields

$$\dot{s} = \frac{\partial f}{\partial \delta} \cdot \dot{\delta} \tag{2.7}$$

As $\dfrac{\partial f}{\partial \delta}$ is a known constant defined by the sensor technology used, we can assume without loss of generality that it is equal to 1, by simply normalizing the data.

By construction, we have :

$$\vec{ST} = \delta \, n_S$$

and :

$$\dot{\vec{ST}} = \dot{\delta}\, n_S + \delta\, \dot{n_S} \tag{2.8}$$

Let us express that n_S is linked to F_S:

$$\dot{n_S} = \omega_{F_S/F_T} \times n_S \tag{2.9}$$

Equations 2.8 and 2.9 yield:

$$\dot{\vec{ST}} = \dot{\delta}\, n_S + \delta\, (\omega_{F_S/F_T} \times n_S) \tag{2.10}$$

On the other hand:

$$\dot{\vec{ST}} = V_{T/_{F_T}} - V_{S/_{F_T}} \tag{2.11}$$

Moreover the point T slides on the surface of the target, i.e.:

$$\langle\, n_T \,|\, V_{T/_{F_T}} \,\rangle = 0 \tag{2.12}$$

Substituting $V_{T/_{F_T}}$ in (2.12) and using the properties of the scalar product we obtain:

$$\langle n_T \,|\, \dot{\vec{ST}} \rangle = -\langle n_T \,|\, V_{S/_{F_T}} \rangle \tag{2.13}$$

We can rewrite equations (2.13) and (2.10) as:

$$\langle n_T \,|\, V_{S/_{F_T}} \rangle = -\dot{\delta}\, \langle\, n_T \,|\, n_S \,\rangle - \delta\, \langle\, n_T \,|\, (\omega_{F_S/F_T} \times n_S) \,\rangle \tag{2.14}$$

Finally we obtain:

$$\dot{\delta} = -\frac{1}{\langle\, n_T \,|\, n_S \,\rangle} [\langle\, n_T \,|\, V_{S/_{F_T}} \,\rangle + \delta\, \langle\, (n_T \times n_S) \,|\, \omega_{F_S/F_T} \,\rangle] \tag{2.15}$$

Identifying u and $H(S)$ in the formula $\dot{\delta} = (H(S), u) \bullet T_{ST}$ using the properties of the screw product,we finally obtain the Interaction Screw associated with S and T :

$$\left\{ \begin{array}{lcl} H(S) & = & -\dfrac{\delta}{\langle n_T \,|\, n_S \rangle}\, (n_T \times n_S) \\ \\ u & = & -\dfrac{1}{\langle n_T \,|\, n_S \rangle}\, n_T \end{array} \right. \tag{2.16}$$

This Interaction Screw depends on the geometry of the obstacle by the mean of the normal vector to the surface. Actually this vector is not precisely known a priori, however control schemes can be made robust to overcome this uncertainty.

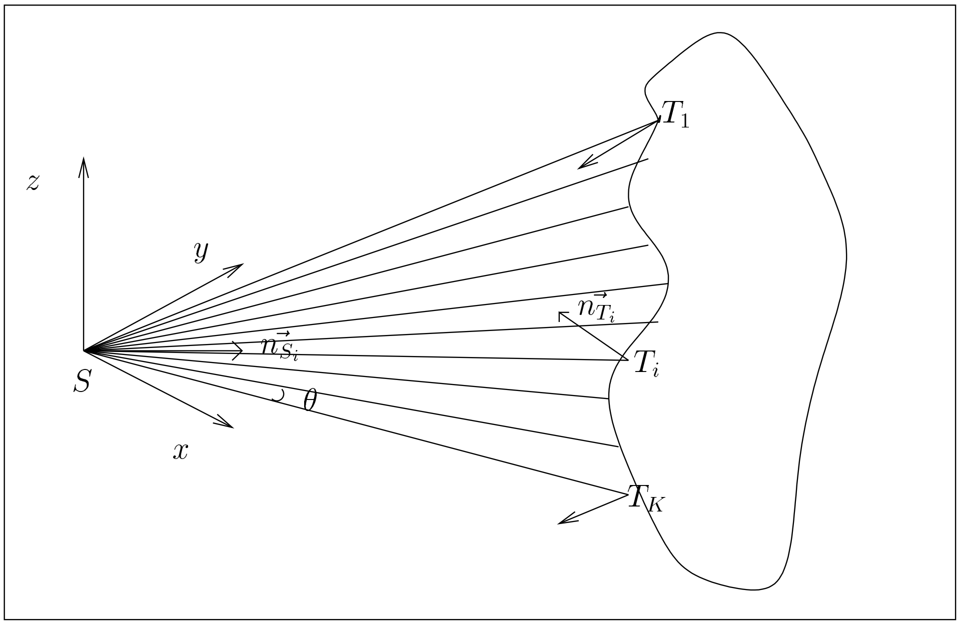
2.3.2 Pin-point Range Finder :

As an acoustic sounder delivers a signal proportional to the distance between the source and the target,we assume than an elementary thin-field range finder can be represented as a pin-point sensor which delivers a signal $s = f(\delta(\vec{n_S}, \vec{n_T}))$ with (Ox) being the line of sight.

Thus we have $\vec{n_S} = (1\ 0\ 0)^T$

2.3.3 Scanside Sonar :

By assembling pin-point sensors lying in a scanning plane with a scanning step angle θ_i we build a more complex model, the scanside sonar.



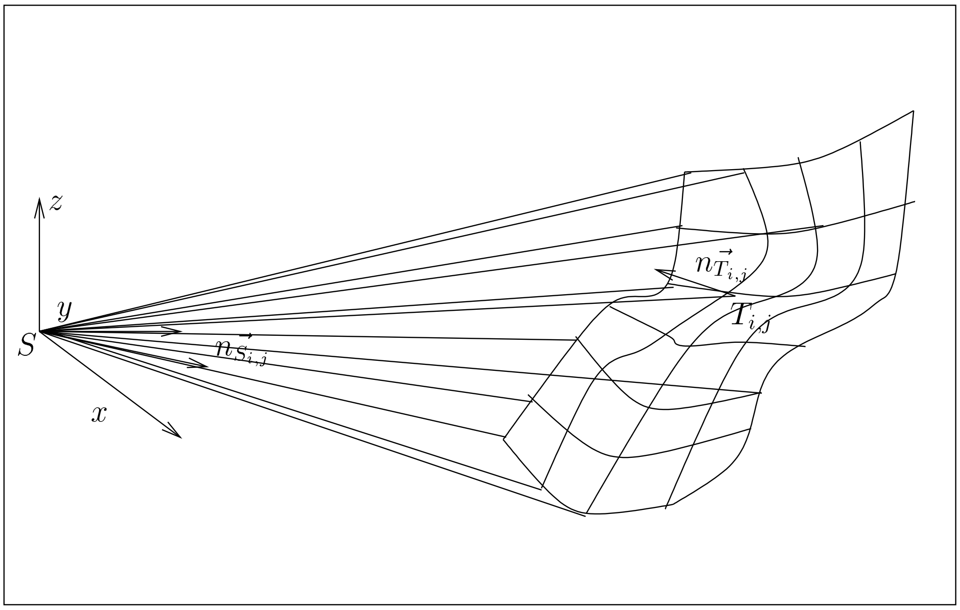
As (Oxy) is the plane of sight we have

$$\begin{aligned} \underline{s} &= (s_1 \ldots s_i \ldots s_K)^T \quad \text{the multidimensional signal} \\ s_i &= f(\delta_i(\vec{n_{S_i}}, \vec{n_{T_i}})) \end{aligned}$$

with

$$\begin{aligned} \delta_i &= ST_i \\ \vec{n_{S_i}} &= (\cos\theta_i\ \sin\theta_i\ 0)^T \end{aligned}$$

2.3.4 3D Camera :



Likewise we can consider a range finder as an assembly of elementary pin-point sensors whose axis are on a regular grid parameterized by pan and tilt angles (θ_i, ϕ_j).

$$\underline{s} = \begin{pmatrix} s_{1,1} & \dots & \dots & \dots & \dots & \dots & s_{1,K} \\ \vdots & & \ddots & & & & \vdots \\ \vdots & & & s_{i,j} & & & \vdots \\ \vdots & & & & \ddots & & \vdots \\ s_{L,1} & \dots & \dots & \dots & \dots & \dots & s_{L,K} \end{pmatrix}$$

$$s_{i,j} = f(\delta_{i,j}(n\vec{S_{i,j}}, n\vec{T_{i,j}}))$$

with

$$\delta_{i,j} = ST_{i,j}$$

$$n\vec{S_{i,j}} = (\cos\theta_i \cos\phi_j \quad \sin\theta_i \cos\phi_j \quad \sin\phi_j)^T$$

This sensor provides a depth map of the scene.

Chapter 3

Determination of the Interaction Matrix for each couple Sensor/3D Feature in the case of range sensors

In the previous chapter we have introduced the interaction screw to characterize the differential of the sensor signal, but in practice we will compute its matricial representation L. All the computations presented here were executed with the symbolic mathematical calculation software *Maple V.3*.

Range sensors only detect surfaces, so we treat only the case of edges, planes, spheres and cylinders. We do not treat the case of 3D points (vertices), which has little physical sense.

We have the following matricial representation of the interaction screw, called *Interaction Matrix*, similarly given by the relation :

$$L_i{}^T = H_i.\begin{pmatrix} 0 & \mathbb{I}_3 \\ \mathbb{I}_3 & 0 \end{pmatrix} \tag{3.1}$$

This means if $H = (H(S), u)$ given by Eq.(2.16), then $L^T = (u, H(S))$.

Thanks to this identification, we can transform the screw product into a matrix product, and then simplify the search of the reciprocical subspace of screws S^* into a computation of the kernel of the matrix L^T. Indeed, thanks to (3.1) we can write Eq. (1.4) with an obvious breach of notation :

$$\dot{s}_i = L^T.T \tag{3.2}$$

thus $\dot{s}_i = 0$ yields $T \in \ker L^T$.

3.1 Pin-point Range Finder :

Let δ be the range provided by an acoustic sounder. Actually n_T should be a unitary vector but as its norm appears in the numerator and in the denominator of the equations. For $H(S)$ and u, we will not normalize n_T in order to avoid spreading numerical errors.

3.1.1 Plane :

As $n_T = (a\ b\ c)^T$ we obtain the following interaction matrix:

$$L^T = \begin{pmatrix} -1 & -\frac{b}{a} & -\frac{c}{a} & 0 & -\frac{\delta c}{a} & \frac{\delta b}{a} \end{pmatrix} \tag{3.3}$$

The free space basis is :

$$S^* = \begin{pmatrix} -c & -b & -\delta c & \delta b & 0 \\ 0 & a & 0 & 0 & 0 \\ a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & a & 0 & 0 \\ 0 & 0 & 0 & a & 0 \end{pmatrix} \tag{3.4}$$

3.1.2 Edge :

In the case of an edge we consider the normal vector $(a_m\ b_m\ c_m)^T$ of each of the two planes defining the edge to get two interaction sub-matrices that we will stack together to obtain the interaction matrix :

$$L^T = \begin{pmatrix} -1 & -\frac{b_1}{a_1} & -\frac{c_1}{a_1} & 0 & -\frac{\delta c_1}{a_1} & \frac{\delta b_1}{a_1} \\ -1 & -\frac{b_2}{a_2} & -\frac{c_2}{a_2} & 0 & -\frac{\delta c_2}{a_2} & \frac{\delta b_2}{a_2} \end{pmatrix} \tag{3.5}$$

And then the free space basis is:

$$S^* = \ker L^T = \begin{pmatrix} c_1\,b_2 - c_2\,b_1 & \delta(b_1\,c_2 - b_2\,c_1) & 0 & 0 \\ a_1\,c_2 - a_2\,c_1 & \delta(a_2\,c_1 - a_1\,c_2) & \delta & 0 \\ a_2\,b_1 - a_1\,b_2 & 0 & 0 & 0 \\ a_2\,b_1 - a_1\,b_2 & 0 & 0 & 1 \\ 0 & a_1\,b_2 - a_2\,b_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{3.6}$$

3.1.3 Sphere

Let us consider a sphere of center $\mathcal{O} = (x_0, y_0, z_0)$ and with radius R.
Since T is the intersection point of the sensor axis and of the feature, its coordinate are given by $ST = \delta n_S$.
We have:

$$n_T = k.\nabla\mathcal{F}.(\delta n_S)$$

Here $\mathcal{F}(x,y,z) = (x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2 - R^2$.
Thus:

$$\nabla\mathcal{F}(x,y,z) = 2\begin{pmatrix} x-x_0 \\ y-y_0 \\ z-z_0 \end{pmatrix}$$

As we consider only the direction of n_T we can write:

$$n_T = (\delta{-}x_0 \;\; -y_0 \;\; -z_0)^T$$

So we obtain the interaction matrix:

$$L^T = \begin{pmatrix} -1 & -\dfrac{y_0}{x_0-\delta} & -\dfrac{z_0}{x_0-\delta} & 0 & -\dfrac{\delta z_0}{x_0-\delta} & \dfrac{\delta y_0}{x_0-\delta} \end{pmatrix} \tag{3.7}$$

And the free space basis is:

$$S^* = \begin{pmatrix} 0 & y_0 & 0 & 0 & 0 \\ -z_0 & \delta-x_0 & \delta & 0 & -\delta\, z_0 \\ y_0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & y_0 & 0 & y_0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{3.8}$$

3.1.4 Cylinder :

Let us consider a cylinder of radius R and with generating axis $D(M, \vec{v})$ where $M = (x_0\, y_0\, z_0)^T$ is the point of the axis belonging to the plane passing through the origin and of normal axis $\vec{v} = (a\, b\, c)^T$, with $\|\vec{v}\| = 1$.

The method to compute the interaction matrix is very similar to the sphere case.
Here $\mathcal{F}(x,y,z) = (x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2 - [a(x-x_0) + b(y-y_0) + c(z-z_0)]^2 - R^2$.
Thus:

$$\nabla\mathcal{F}(x,y,z) = \begin{pmatrix} 2(x-x_0) - a[a(x-x_0) + b(y-y_0) + c(z-z_0)] \\ 2(y-y_0) - b[a(x-x_0) + b(y-y_0) + c(z-z_0)] \\ 2(z-z_0) - c[a(x-x_0) + b(y-y_0) + c(z-z_0)] \end{pmatrix}$$

Applying $\nabla\mathcal{F}$ at $(\delta,0,0)$ we obtain n_T.
And finally:

$$L = \begin{pmatrix} -1 \\ -\dfrac{y_0\,b^2 + y_0 + z_0\,bc - ab\delta + abx_0}{b^2\delta - b^2x_0 - 2\,\delta + 2\,x_0 + c^2\delta - c^2x_0 + z_0\,ac + y_0\,ab} \\ -\dfrac{z_0 + z_0\,c^2 - ac\delta + acx_0 + y_0\,bc}{b^2\delta - b^2x_0 - 2\,\delta + 2\,x_0 + c^2\delta - c^2x_0 + z_0\,ac + y_0\,ab} \\ 0 \\ -\dfrac{\delta\left(z_0 + z_0\,c^2 - ac\delta + acx_0 + y_0\,bc\right)}{b^2\delta - b^2x_0 - 2\,\delta + 2\,x_0 + c^2\delta - c^2x_0 + z_0\,ac + y_0\,ab} \\ \dfrac{\delta\left(y_0\,b^2 + y_0 + z_0\,bc - ab\delta + abx_0\right)}{b^2\delta - b^2x_0 - 2\,\delta + 2\,x_0 + c^2\delta - c^2x_0 + z_0\,ac + y_0\,ab} \end{pmatrix} \tag{3.9}$$

The free space basis is:

$$S^* = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ l_1 & l_2 & 0 & l_3 & \delta \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{3.10}$$

where:

$$\begin{aligned} l_1 &= \frac{2\,\delta - 2\,x_0 + (-\delta + x_0)\,c^2 - z_0\,ac + (-\delta + x_0)\,b^2 - y_0\,ab}{y_0\,b^2 + y_0 + z_0\,bc + (-\delta + x_0)\,ab} \\ l_2 &= \frac{-y_0\,bc - z_0 - z_0\,c^2 + (\delta - x_0)\,ca}{y_0\,b^2 + y_0 + z_0\,bc + (-\delta + x_0)\,ab} \\ l_3 &= \frac{-\delta y_0\,bc - \delta z_0 - \delta z_0\,c^2 - \delta\,(-\delta + x_0)\,ca}{y_0\,b^2 + y_0 + z_0\,bc + (-\delta + x_0)\,ab} \end{aligned}$$

3.2 Scanside Sonar :

3.2.1 Plane :

We apply the results of section 2.1.1 with:

$$\begin{cases} n_{S_i} = (\cos\theta_i\ \ \sin\theta_i\ \ 0)^T \\ n_{T_i} = (a\ b\ c)^T \end{cases}$$

We obtain the interaction matrix $(L_i)_{i=1..K}$ with:

$$L_i = \begin{pmatrix} -\dfrac{a}{\cos\theta_i a + \sin\theta_i b} \\ -\dfrac{b}{\cos\theta_i a + \sin\theta_i b} \\ -\dfrac{c}{\cos\theta_i a + \sin\theta_i b} \\ \dfrac{\delta_i c \sin\theta_i}{\cos\theta_i a + \sin\theta_i b} \\ -\dfrac{\delta_i c \cos\theta_i}{\cos\theta_i a + \sin\theta_i b} \\ \dfrac{\delta_i\left(-a\sin\theta_i + b\cos\theta_i\right)}{\cos\theta_i a + \sin\theta_i b} \end{pmatrix} \tag{3.11}$$

And the free space basis is:

$$S^* = \begin{pmatrix} -c & -b & 0 \\ 0 & a & 0 \\ a & 0 & 0 \\ 0 & 0 & a \\ 0 & 0 & b \\ 0 & 0 & c \end{pmatrix} \tag{3.12}$$

3.2.2 Edge :

The interaction matrix is composed of the elementary matrices:

$$L_i = \begin{pmatrix} -\dfrac{a_1}{\cos\theta_i a_1 + \sin\theta_i b_1} & -\dfrac{a_2}{\cos\theta_i a_2 + \sin\theta_i b_2} \\ -\dfrac{b_1}{\cos\theta_i a_1 + \sin\theta_i b_1} & -\dfrac{b_2}{\cos\theta_i a_2 + \sin\theta_i b_2} \\ -\dfrac{c_1}{\cos\theta_i a_1 + \sin\theta_i b_1} & -\dfrac{c_2}{\cos\theta_i a_2 + \sin\theta_i b_2} \\ \dfrac{\delta_i c_1 \ \sin\theta_i}{\cos\theta_i a_1 + \sin\theta_i b_1} & \dfrac{\delta_i c_2 \ \sin\theta_i}{\cos\theta_i a_2 + \sin\theta_i b_2} \\ -\dfrac{\delta_i c_1 \ \cos\theta_i}{\cos\theta_i a_1 + \sin\theta_i b_1} & -\dfrac{\delta_i c_2 \ \cos\theta_i}{\cos\theta_i a_2 + \sin\theta_i b_2} \\ -\dfrac{\delta_i(a_1 \ \sin\theta_i - b_1 \ \cos\theta_i)}{\cos\theta_i a_1 + \sin\theta_i b_1} & -\dfrac{\delta_i(a_2 \ \sin\theta_i - b_2 \ \cos\theta_i)}{\cos\theta_i a_2 + \sin\theta_i b_2} \end{pmatrix} \tag{3.13}$$

The free space basis is:

$$S^* = \begin{pmatrix} b_1\,c_2 - c_1\,b_2 \\ c_1\,a_2 - a_1\,c_2 \\ b_2\,a_1 - a_2\,b_1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{3.14}$$

3.2.3 Sphere :

The normal vector is obtained in the same way as for an pin-point range finder, with $ST_i = \delta_i.n_{s_i}$. So the interaction matrix is composed of the elementary matrices:

$$L_i = \begin{pmatrix} -\dfrac{-\delta_i \cos\theta_i + x_0}{\cos\theta_i x_0 - \delta_i + \sin\theta_i y_0} \\ -\dfrac{-\delta_i \sin\theta_i + y_0}{\cos\theta_i x_0 - \delta_i + \sin\theta_i y_0} \\ -\dfrac{z_0}{\cos\theta_i x_0 - \delta_i + \sin(\theta_i) y_0} \\ \dfrac{\delta_i z_0\ \sin\theta_i}{\cos\theta_i x_0 - \delta_i + \sin\theta_i y_0} \\ -\dfrac{\delta_i z_0\ \cos\theta_i}{\cos\theta_i x_0 - \delta_i + \sin\theta_i y_0} \\ -\dfrac{\delta_i\,(x_0\ \sin\theta_i - \cos\theta_i y_0)}{\cos\theta_i x_0 - \delta_i + \sin\theta_i y_0} \end{pmatrix} \tag{3.15}$$

And the free space basis is:

$$S^* = \begin{pmatrix} 0 & -y_0 & z_0 \\ -z_0 & x_0 & 0 \\ y_0 & 0 & -x_0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \tag{3.16}$$

3.2.4 Cylinder :

By analogy we obtain the interaction matrix by stacking the elementary matrices :

$$L_i = \begin{pmatrix} -\dfrac{2\,\delta_i\cos\theta_i - \delta_i\cos\theta_i b^2 + ab\delta_i\sin\theta_i - \delta_i\cos\theta_i c^2 - x_0}{\delta_i\cos^2\theta_i - 2\,\delta_i\cos^2\theta_i b^2 + 2\,\cos\theta_i ab\delta_i\sin\theta_i - \delta_i\cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0} \\ -\dfrac{\delta_i\sin\theta_i + b^2\delta_i\sin\theta_i + ba\delta_i\cos\theta_i - y_0}{\delta_i\cos^2\theta_i - 2\,\delta_i\cos^2\theta_i b^2 + 2\,\cos\theta_i ab\delta_i\sin\theta_i - \delta_i\cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0} \\ -\dfrac{-z_0 + ca\delta_i\cos\theta_i + cb\delta_i\sin\theta_i}{\delta_i\cos^2\theta_i - 2\,\delta_i\cos^2\theta_i b^2 + 2\,\cos\theta_i ab\delta_i\sin\theta_i - \delta_i\cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0} \\ \dfrac{\delta_i\left(-z_0\,\sin\theta_i + ca\delta_i\cos\theta_i\sin\theta_i + cb\delta_i - cb\delta_i\cos^2\theta_i\right)}{\delta_i\cos^2\theta_i - 2\,\delta_i\cos^2\theta_i b^2 + 2\,\cos\theta_i ab\delta_i\sin\theta_i - \delta_i\cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0} \\ -\dfrac{\delta_i(-z_0 + ca\delta_i\cos\theta_i + cb\delta_i\sin\theta_i)\cos\theta_i}{\delta_i\cos^2\theta_i - 2\,\delta_i\cos^2\theta_i b^2 + 2\,\cos\theta_i ab\delta_i\sin\theta_i - \delta_i\cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0} \\ \dfrac{\delta_i\left(-\cos\theta_i\delta_i\sin\theta_i + 2\,\delta_i\cos\theta_i b^2\sin\theta_i - ab\delta_i + 2\,ab\delta_i\cos^2\theta_i + \sin\theta_i\delta_i\cos\theta_i c^2 + \sin\theta_i x_0 - y_0\,\cos\theta_i\right)}{\delta_i\cos^2\theta_i - 2\,\delta_i(\cos\theta_i)^2 b^2 + 2\,\cos\theta_i ab\delta_i\sin\theta_i - \delta_i\cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0} \end{pmatrix} \tag{3.17}$$

And the free space basis is:

$$S^* = \begin{pmatrix} a & c\,y_0 - b\,z_0 & 0 & 0 \\ b & a\,z_0 - c\,x_0 & 0 & 0 \\ c & b\,x_0 - a\,y_0 & 0 & 0 \\ 0 & a & 1 & 0 \\ 0 & b & 0 & 1 \\ 0 & c & 0 & 0 \end{pmatrix} \tag{3.18}$$

3.3 3D Camera :

3.3.1 Plane :

Likewise in section 2.1.1. we obtain as :

$$\begin{cases} n_{S_{i,j}} = (\cos\theta_i\ \cos\phi_j\ \ \sin\theta_i\ \cos\phi_j\ \ \sin\phi_j)^T \\ n_{T_{i,j}} = (a\ b\ c)^T \end{cases}$$

the interaction matrix $(H_{i,j})_{i=1..K,j=1..L}$, with:

$$H_{i,j} = \begin{pmatrix} -\dfrac{a}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \\ -\dfrac{b}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \\ -\dfrac{c}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \\ \dfrac{\delta_{i,j}(-\sin\theta_i\cos\phi_j c + \sin\phi_j b)}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \\ -\dfrac{\delta_{i,j}(a\sin\phi_j - c\cos\theta_i\cos\phi_j)}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \\ \dfrac{\delta_{i,j}\cos\phi_j(a\sin(\theta_i - b\cos\theta_i))}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \end{pmatrix} \tag{3.19}$$

And the free space basis is:

$$S^* = \begin{pmatrix} 0 & a & 0 \\ c & 0 & 0 \\ -b & -c & 0 \\ 0 & 0 & a \\ 0 & 0 & b \\ 0 & 0 & c \end{pmatrix} \tag{3.20}$$

3.3.2 Edge :

Likewise again, we have the interaction matrix by stacking the elementary matrices:

$$L_{i,j} = \begin{pmatrix} -\frac{a_1}{a_1\ \cos\theta_i\cos\phi_j + b_1\ \sin\theta_i\cos\phi_j + c_1\ \sin\phi_j} & -\frac{a_2}{a_2\ \cos\theta_i\cos\phi_j + b_2\ \sin\theta_i\cos\phi_j + c_2\ \sin\phi_j} \\ -\frac{b_1}{a_1\ \cos\theta_i\cos\phi_j + b_1\ \sin\theta_i\cos\phi_j + c_1\ \sin\phi_j} & -\frac{b_2}{a_2\ \cos\theta_i\cos\phi_j + b_2\ \sin\theta_i\cos\phi_j + c_2\ \sin\phi_j} \\ -\frac{c_1}{a_1\ \cos\theta_i\cos\phi_j + b_1\ \sin\theta_i\cos\phi_j + c_1\ \sin\phi_j} & -\frac{c_2}{a_2\ \cos\theta_i\cos\phi_j + b_2\ \sin\theta_i\cos\phi_j + c_2\ \sin\phi_j} \\ \frac{\delta_{i,j}(-\sin\theta_i\cos\phi_j c_1 + \sin\phi_j b_1)}{a_1\ \cos\theta_i\cos\phi_j + b_1\ \sin\theta_i\cos\phi_j + c_1\ \sin\phi_j} & \frac{\delta_{i,j}(-\sin\theta_i\cos\phi_j c_2 + \sin\phi_j b_2)}{a_2\ \cos\theta_i\cos\phi_j + b_2\ \sin\theta_i\cos\phi_j + c_2\ \sin\phi_j} \\ -\frac{\delta_{i,j}(\sin\phi_j a_1 - \cos\theta_i\cos\phi_j c_1)}{a_1\ \cos\theta_i\cos\phi_j + b_1\ \sin\theta_i\cos\phi_j + c_1\ \sin\phi_j} & -\frac{\delta_{i,j}(\sin\phi_j a_2 - \cos\theta_i\cos\phi_j c_2)}{a_2\ \cos\theta_i\cos\phi_j + b_2\ \sin\theta_i\cos\phi_j + c_2\ \sin\phi_j} \\ -\frac{\delta_{i,j}\ \cos\phi_j(\cos\theta_i b_1 - \sin\theta_i a_1)}{a_1\ \cos\theta_i\cos\phi_j + b_1\ \sin\theta_i\cos\phi_j + c_1\ \sin\phi_j} & \frac{\delta_{i,j}\ \cos\phi_j(-\cos\theta_i b_2 + \sin\theta_i a_2)}{a_2\ \cos\theta_i\cos\phi_j + b_2\ \sin\theta_i\cos\phi_j + c_2\ \sin\phi_j} \end{pmatrix} \tag{3.21}$$

And the free space basis is:

$$S^* = \begin{pmatrix} c_2\,b1 - b_2\,c_1 \\ a_2\,c_1 - c_2\,a_1 \\ b_2\,a_1 - b_1\,a_2 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{3.22}$$

3.3.3 Sphere :

By the same way as for an pin-point range finder or a scanning laser range finder, we compute $n_{T_{i,j}}$ thanks to $ST_{i,j} = \delta_{i,j}.n_{s_{i,j}}$ and we obtain the elementary interaction matrix:

$$H_{i,j} = \begin{pmatrix} \frac{\delta_{i,j}\cos\theta_i\cos\phi_j - x_0}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \\ \frac{\delta_{i,j}\sin\theta_i\cos\phi_j - y_0}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \\ -\frac{-\delta_{i,j}\sin\phi_j + z_0}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \\ -\frac{\delta_{i,j}\left(\sin\theta_i\cos\phi_j z_0 - \sin\phi_j y_0\right)}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \\ \frac{\delta_{i,j}\left(-\sin\phi_j x_0 + \cos\theta_i\cos\phi_j z_0\right)}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \\ -\frac{\cos\phi_j\delta_{i,j}\left(\cos\theta_i y_0 - \sin\theta_i x_0\right)}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \end{pmatrix} \tag{3.23}$$

And the free space basis is:

$$S^* = \begin{pmatrix} 0 & y_0 & z_0 \\ z_0 & -x_0 & 0 \\ -y_0 & 0 & x_0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \tag{3.24}$$

3.3.4 Cylinder :

Following the same method, we find the interaction matrix :

$$L_{i,j} = \frac{1}{e} l_{i,j} \tag{3.25}$$

where:

$$\begin{aligned} e \;=\; & cos^2\theta_i \cos^2 phi_j \delta_{i,j} - cos\theta_i \cos\phi_j x_0 + 2\cos\theta_i\cos\phi_j ac\delta_{i,j}\sin\phi_j - 2\cos^2\theta_i\cos^2\phi_j\delta_{i,j}b^2 \\ & +2\cos\theta_i\cos^2\phi_j ab\delta_{i,j}\sin\theta_i - \cos^2\theta_i\cos^2\phi_j\delta_{i,j}c^2 - \sin\theta_i\cos\phi_j y_0 + \cos^2\phi_j b^2\delta_{i,j} \\ & +2\sin\theta_i\cos\phi_j bc\delta_{i,j}\sin\phi_j + \delta_{i,j} - \sin\phi_j z_0 + c^2\delta_{i,j} - c^2\delta_{i,j}\cos^2\phi_j \end{aligned}$$

and:

$$
\begin{aligned}
l_{i,j}(1) &= -2\,\delta_{i,j}\cos\theta_i\cos\phi_j + x_0 - ac\delta_{i,j}\sin\phi_j + \delta_{i,j}\cos\theta_i\cos\phi_j b^2 \\
&\quad -ab\delta_{i,j}\sin\theta_i\cos\phi_j + \delta_{i,j}\cos\theta_i\cos\phi_j c^2 \\
l_{i,j}(2) &= -\delta_{i,j}\sin\theta_i\cos\phi_j + y_0 - b^2\delta_{i,j}\sin\theta_i\cos\phi_j - bc\delta_{i,j}\sin\phi_j - ba\delta_{i,j}\cos\theta_i\cos\phi_j \\
l_{i,j}(3) &= -\delta_{i,j}\sin\phi_j + z_0 - ca\delta_{i,j}\cos\theta_i\cos\phi_j - cb\delta_{i,j}\sin\theta_i\cos\phi_j - c^2\delta_{i,j}\sin\phi_j \\
l_{i,j}(4) &= \delta_{i,j}(\sin\theta_i\cos\phi_j z_0 - \sin\theta_i\cos^2\phi_j ca\delta_{i,j}\cos\theta_i - 2\,\cos^2\phi_j cb\delta_{i,j} + \cos^2\phi_j cb\delta_{i,j}\cos^2\theta_i \\
&\quad -\sin\theta_i\cos\phi_j c^2\delta_{i,j}\sin\phi_j - \sin\phi_j y_0 + \sin\phi_j b^2\delta_{i,j}\sin\theta_i\cos\phi_j + bc\delta_{i,j} + \sin\phi_j ba\delta_{i,j}\cos\theta_i\cos\phi_j) \\
l_{i,j}(5) &= -\delta_{i,j}(\sin\phi_j\delta_{i,j}\cos\theta_i\cos\phi_j - \sin\phi_j x_0 + ac\delta_{i,j} - ac\delta_{i,j}\cos^2\phi_j - \sin\phi_j\delta_{i,j}\cos\theta_i\cos\phi_j b^2 \\
&\quad +\sin\phi_j ab\delta_{i,j}\sin\theta_i\cos\phi_j - 2\,\sin\phi_j\delta_{i,j}\cos\theta_i\cos\phi_j c^2 + \cos\theta_i\cos\phi_j z_0 - \cos^2\theta_i\cos^2\phi_j ca\delta_{i,j} \\
&\quad -\cos\theta_i\cos^2 cb\delta_{i,j}\sin\theta_i) \\
l_{i,j}(6) &= -\delta_{i,j}\cos\phi_j(-\cos\theta_i\delta_{i,j}\sin\theta_i\cos\phi_j - \cos\theta_i y_0 + 2\,\cos\theta_i b^2\delta_{i,j}\sin\theta_i\cos\phi_j + \cos\theta_i bc\delta_{i,j}\sin\phi_j \\
&\quad +2\,ba\delta_{i,j}\cos^2\theta_i\cos\phi_j + \sin\theta_i x_0 - \sin\theta_i ac\delta_{i,j}\sin\phi_j - \delta_{i,j}\cos\phi_j ab + \sin\theta_i\delta_{i,j}\cos\theta_i\cos\phi_j c^2)
\end{aligned}
$$

And the free space basis is:

$$
S^* = \begin{pmatrix} a & c\,y_0 - b\,z_0 \\ b & a\,z_0 - c\,x_0 \\ c & b\,x_0 - a\,y_0 \\ 0 & a \\ 0 & b \\ 0 & c \end{pmatrix} \tag{3.26}
$$

Chapter 4

Kinematic constraints and linkages

In the previous chapters we have determined how to compute interaction screws from our models of the sensors. We now show that the properties of the interaction screw can be analyzed in terms of constraints in the configuration space, and represented by a formalism close to the theory of kinematics of contacts between solids, the *virtual linkages approach.*

We then explain how to use sensor signals for the design of sensor-based tasks at a task planning level, introducing elementary sensor-based primitives and specifying more elaborate tasks by the mean of multi-sensor fusion.

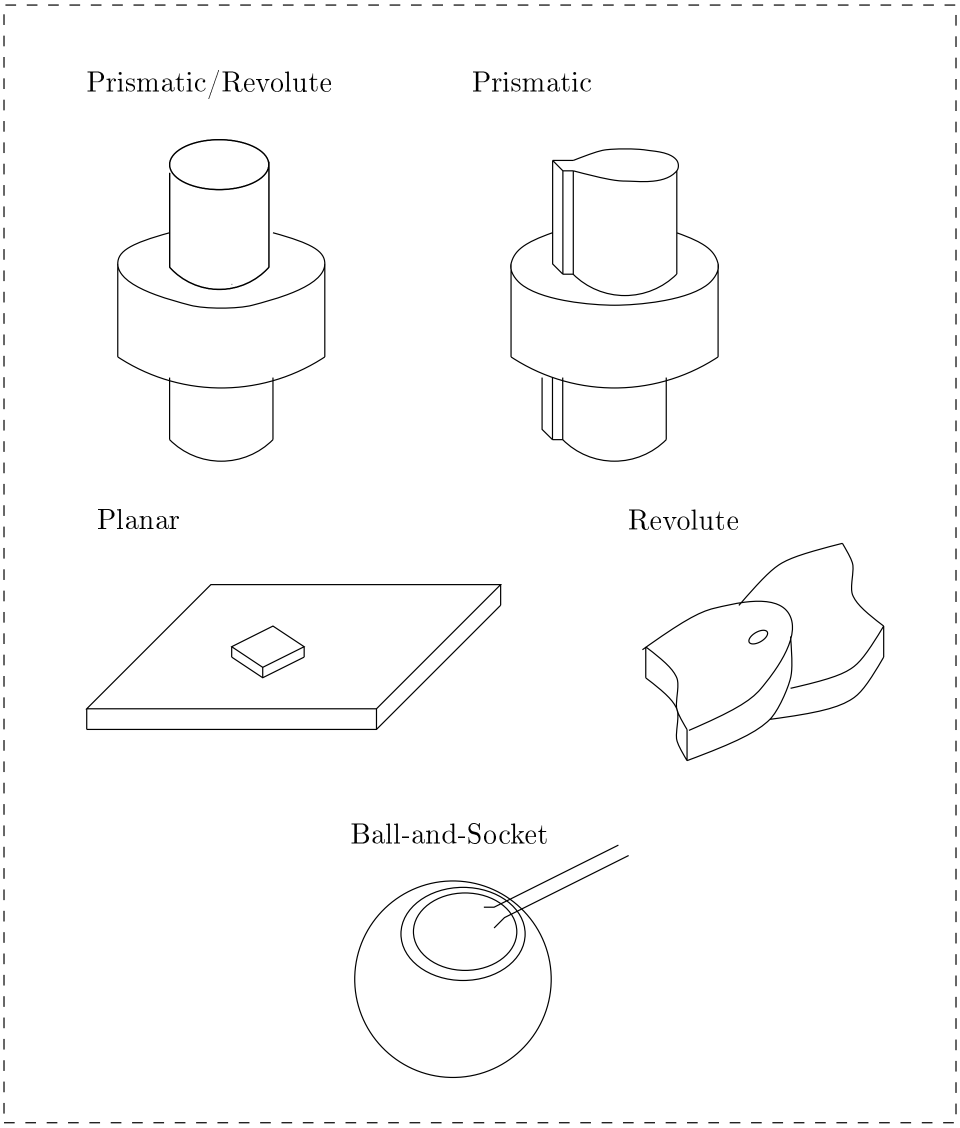
4.1 Virtual linkages

4.1.1 What is a virtual linkage

We now know how to express the differential of the sensor signal s_i as a screw product (Eq. 2.4).The null-space of this differential can be interpreted as a set of displacements ${T^*}_{ST}$ around a situation $\underline{r}$ of the sensor with respect to the target leaving the sensor signal invariant. But as the formalism of screws is a straightforward tool to describe contacts between objects, it introduces a strong analogy between sensor-based tasks and motions with contact. The underlying idea is the following: imposing $\underline{\dot{s}} = 0$ is equivalent to introduce constraints in the configuration space SE_3, as if the target and the sensor were linked geometrically .

A set of compatible constraints $\underline{s}(\underline{r}) - \underline{s}_d = 0$ determines a *virtual linkage* between the sensor S and the target T.

These linkages can be illustrated by:



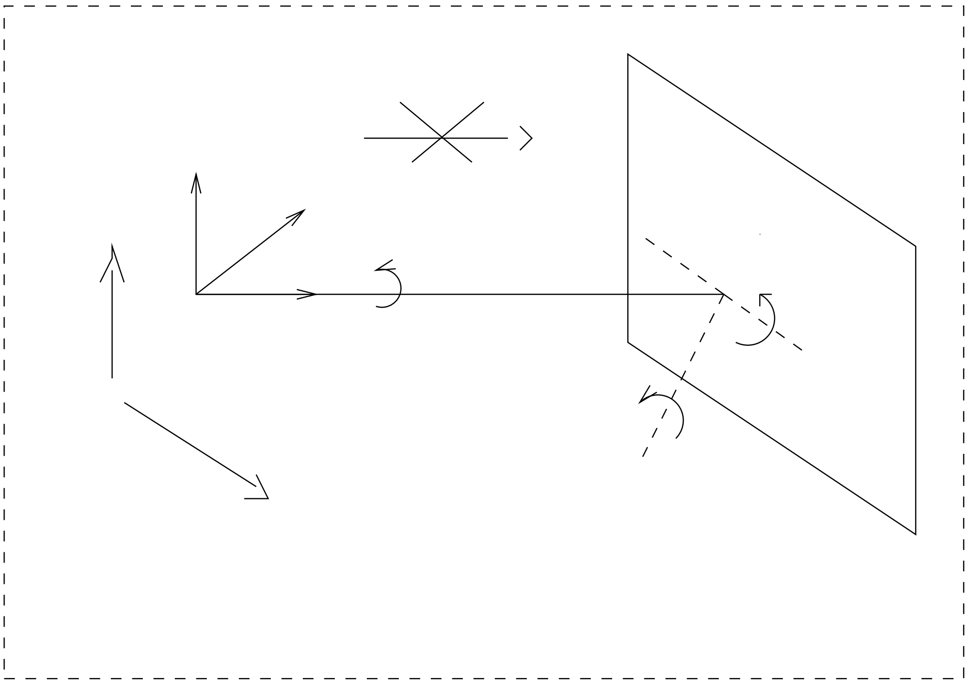
The displacements ${T^*}_{ST}$ computed in the previous chapter will then be called "free motions": the subspace S^* represents the "free motion space" in the case of a *virtual contact* between the sensor S and the target T. For instance, in the case of an thin-field range finder, one can illustrate the constraint $ST = \delta_d$, with δ_d the desired constant range measure, by imaginating a rigid bar linking S and T. In the case of a laser scanning telemeter, it would be a set of rigid bars of different lengths in a same plane of sight linking some points T_i over the surface of the target, and in the case of a range finder the rigid bars linking S to $T_{i,j}$ could be contained in different planes.

At a location $\underline{r}$ where the constraints are satisfied, the dimension N of the reciprocical subspace S^* in se_3 is called the *class* of the virtual linkage at $\underline{r}$. The value of N is directly given by the rank of L^T.

Keeping on the analogy, S^* gives clearly the number of degrees of freedom (D.O.F) of the sensor corresponding to the virtual contact considered. Even better, it allows us to have a geometric interpretation of these families of free motions. With a breach of language, we will assimilate a family of translations with one parameter with a translation. We do the same for the rotations.

4.1.2 Geometric interpretation of the S^* computed for the couples Range Sensor/3D Feature

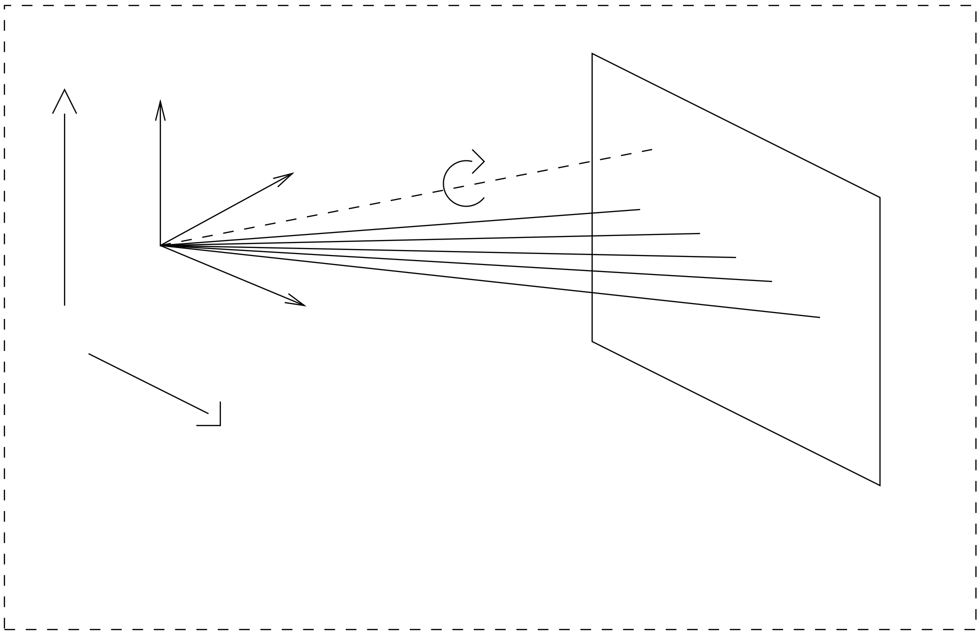
3D Planes



In the case of a thin-field range finder detecting a plane, it is obvious that the only family of constrained motions is a translation parallel to the normal axis of the plane. We see from Equation (3.4) that the families of free motions are :

- the two translations parallel to the plane
- the two families of rotations around the axes of the plane passing by the point T
- the rotation around the axis (ST)

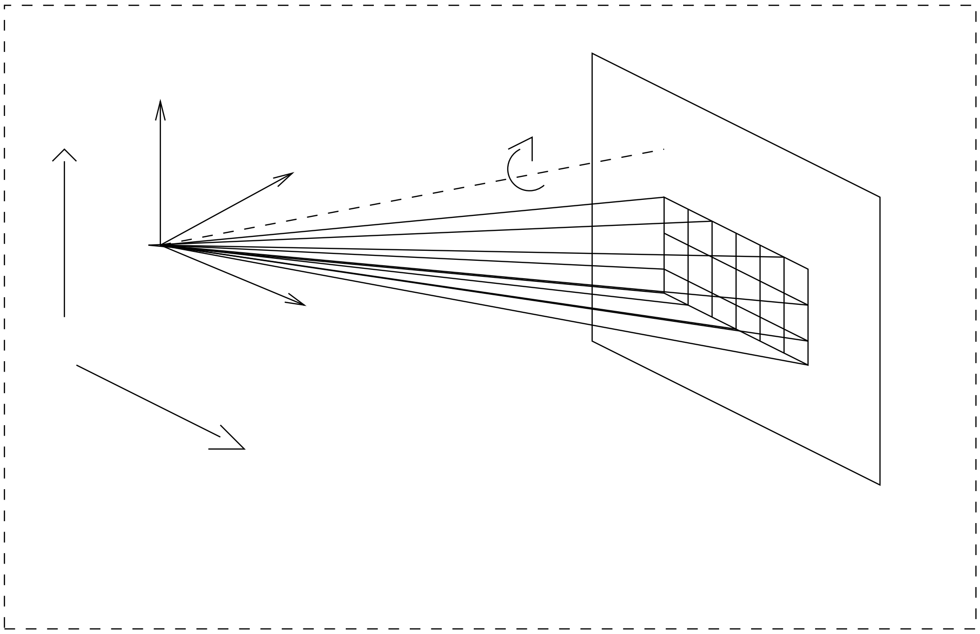
We immediately see that the class of this linkage is 5.It is called a Point/Plane linkage.



In the case of a scanning telemeter, the free motions are:

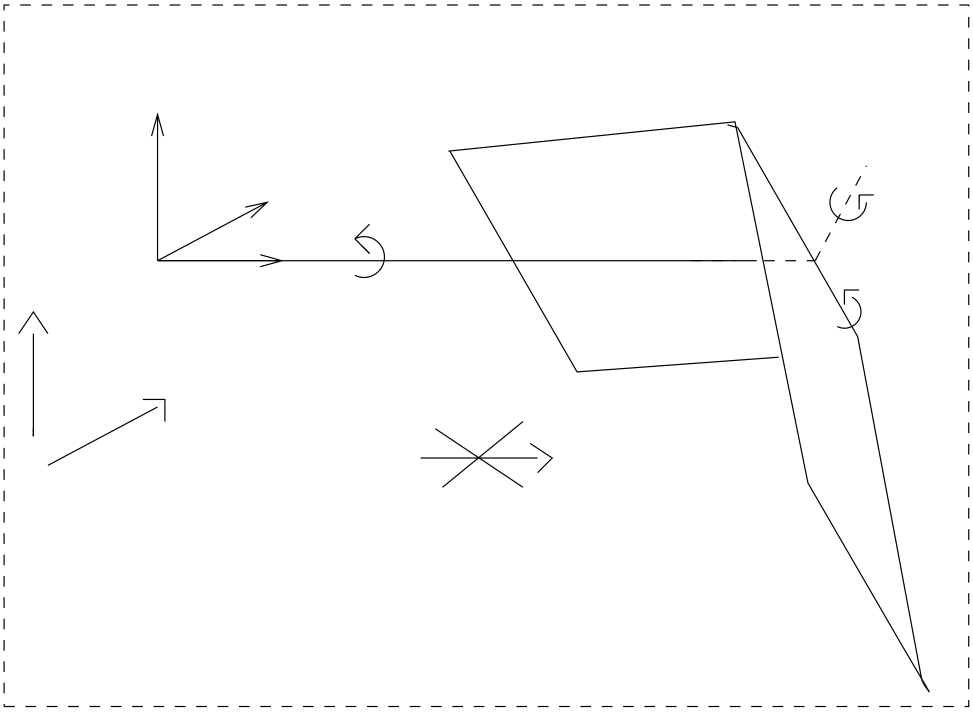
- the two translations parallel to the plane
- the rotation around the normal axis to the plane passing by S

The class of this linkage is 3. It is called a Plane/Plane linkage. Similarly for a 3D camera:



We have the same free motions.

3D Edges

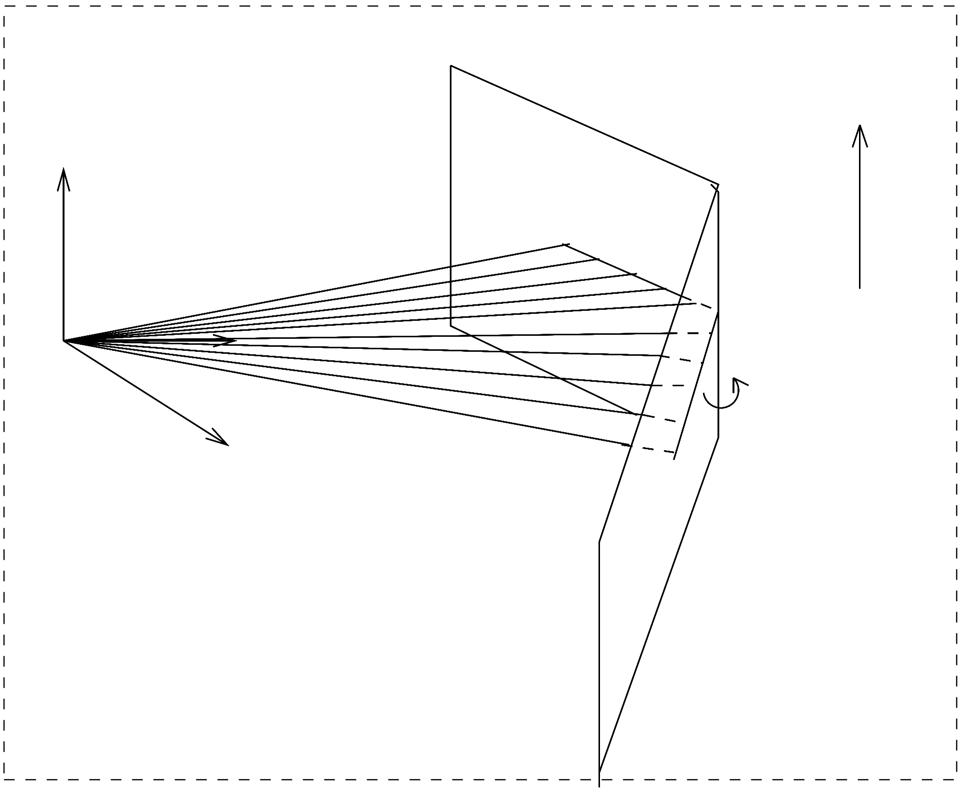


For a thin-field range finder detecting an edge the only constrained motions are the two translations parallel to the normal axis of each of the two planes defining the edge. And the free motions are :

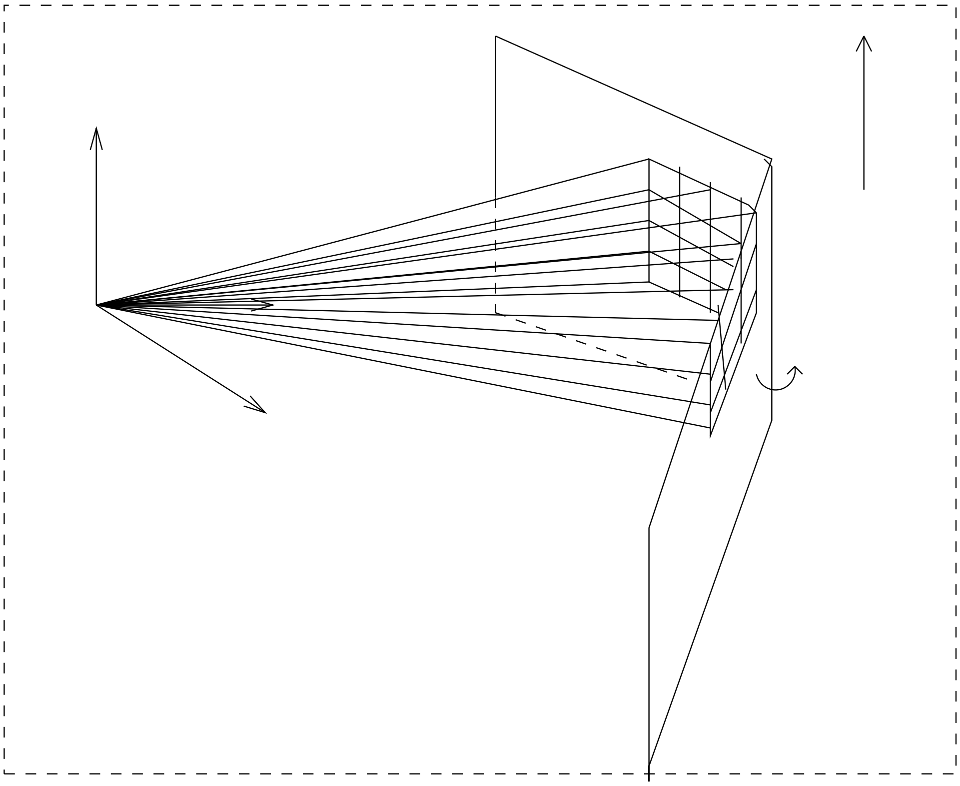
- the translation parallel to the edge
- the rotation around the edge
- the rotation around the axis parallel to (Oz) passing by T
- the rotation around the axis of sight of the sensor

The class of this linkage is 4. It is called a Point/Line linkage.

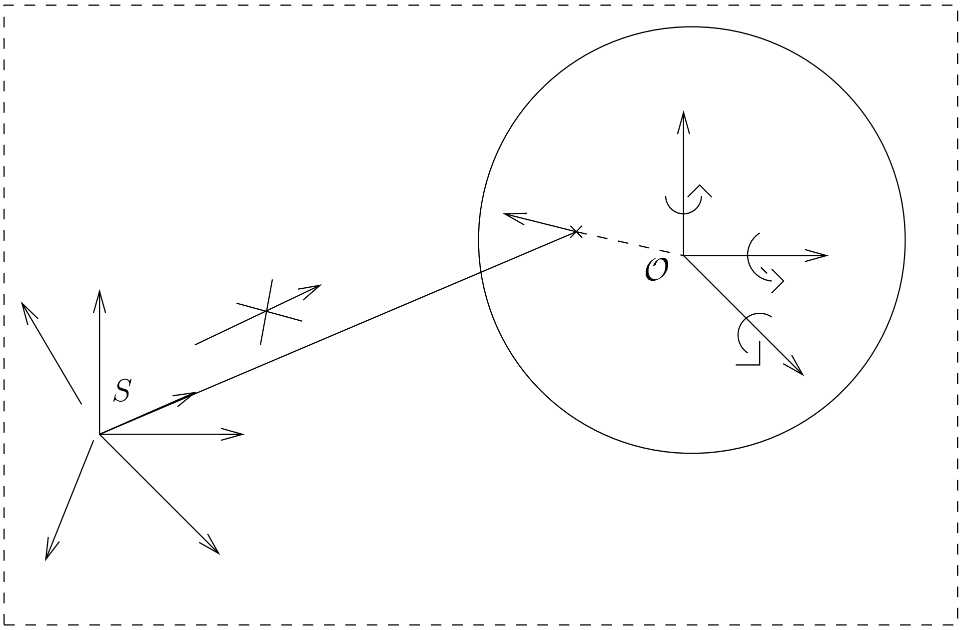
In the case of a scanning telemeter the only free motion is the translation parallel to the edge.It is called a Prismatic linkage, whose class is one.



For a 3D camera the motions left free are the same.



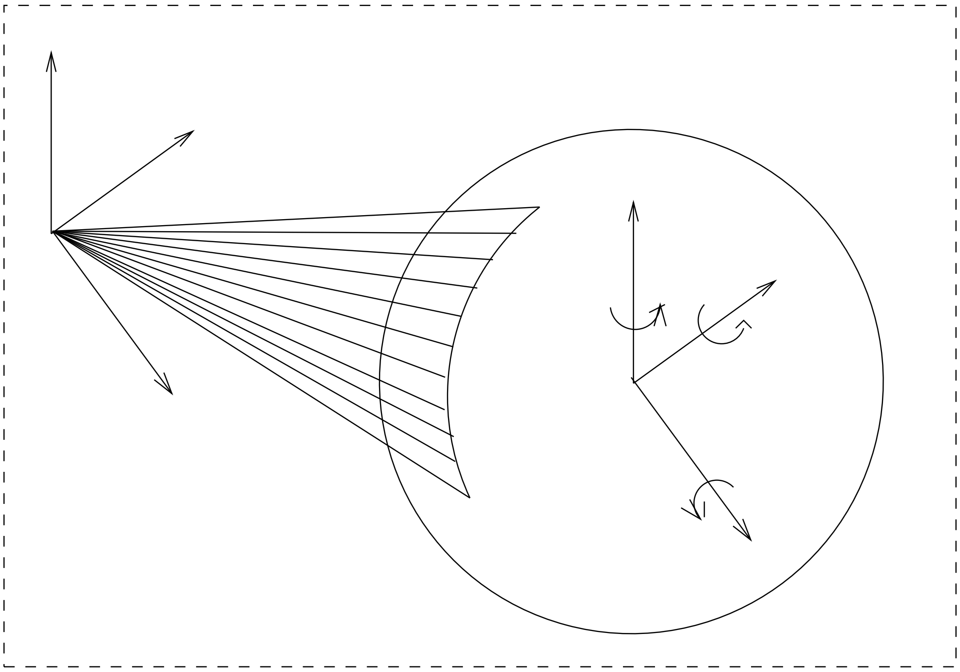
Spheres



For a thin-field range finder detecting a sphere of radius R and of center $\mathcal{O} = (x_0, y_0, z_0)$, the only constrained motion is the translation parallel to $S\mathcal{O}$. And the free motions are :

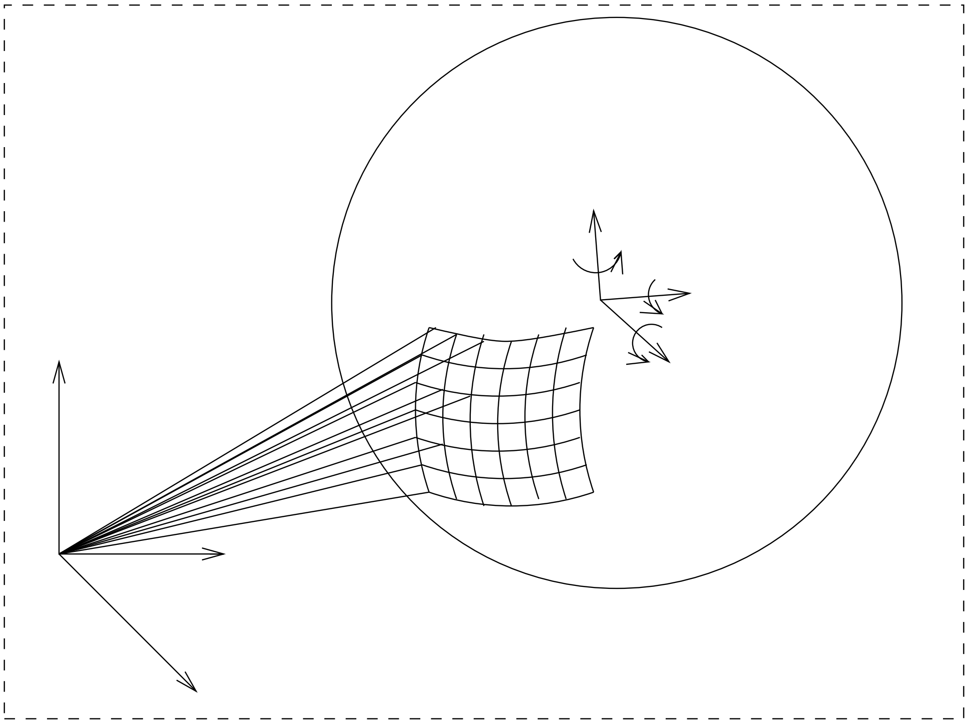
- the two translations orthogonal to $S\mathcal{O}$
- the three rotations around the axes $(\mathcal{O}x), (\mathcal{O}y), (\mathcal{O}z)$

The class of this linkage is 5. It is called a Point/Sphere linkage.

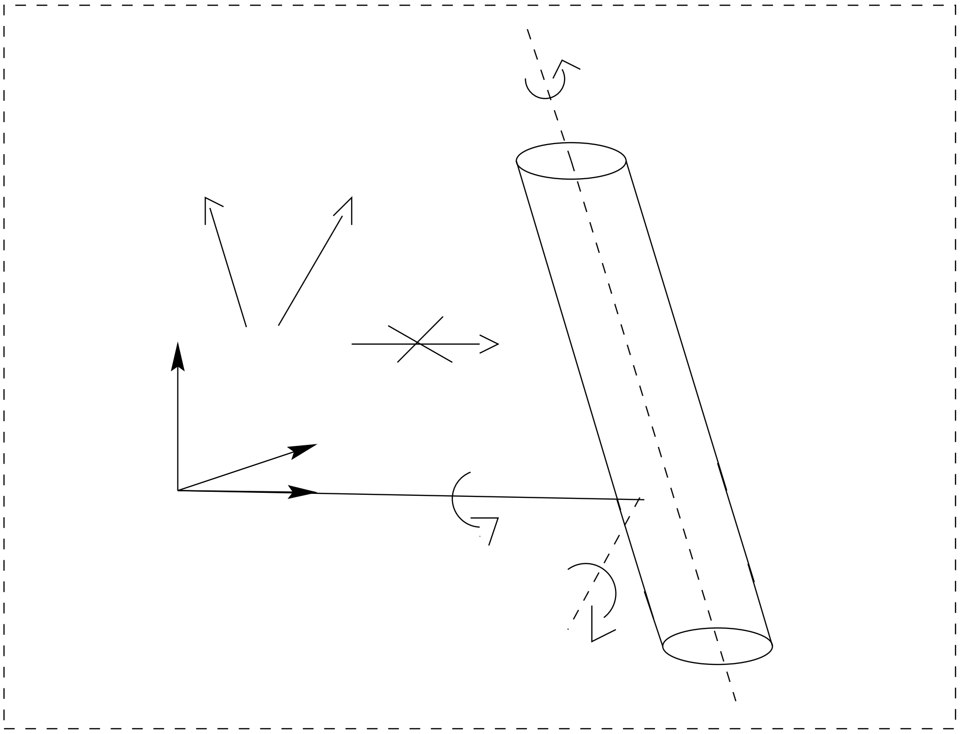


In the case of a scanning telemeter the only free motions are the three rotations around the axes $(\mathcal{O}x), (\mathcal{O}y), (\mathcal{O}z)$. It is called a Ball-and-Socket linkage, whose class is 3.

Similarly for a 3D camera,



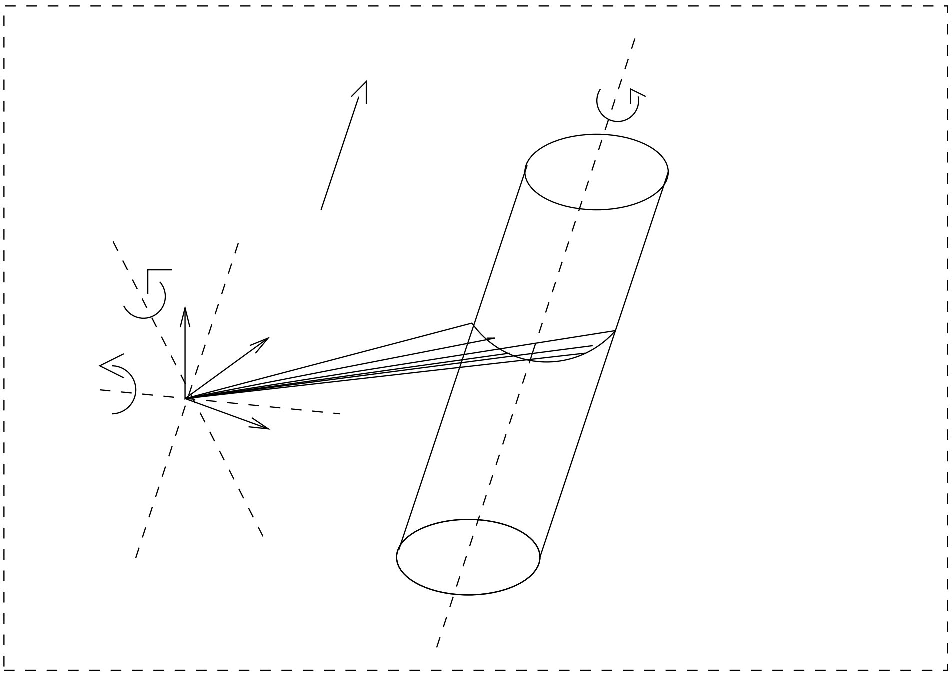
Cylinders



For a thin-field range finder detecting a cylinder the only constrained motion is the translation parallel to the normal axis of the cylinder passing by T. The motions left free are :

- the translation parallel to the generating axis of the cylinder
- the translation orthogonal to this axis and to (ST)
- the rotation around the generating axis of the cylinder
- the rotation around the axis passing by T orthogonal to the axis of the cylinder and to (ST)
- the rotation around (ST)

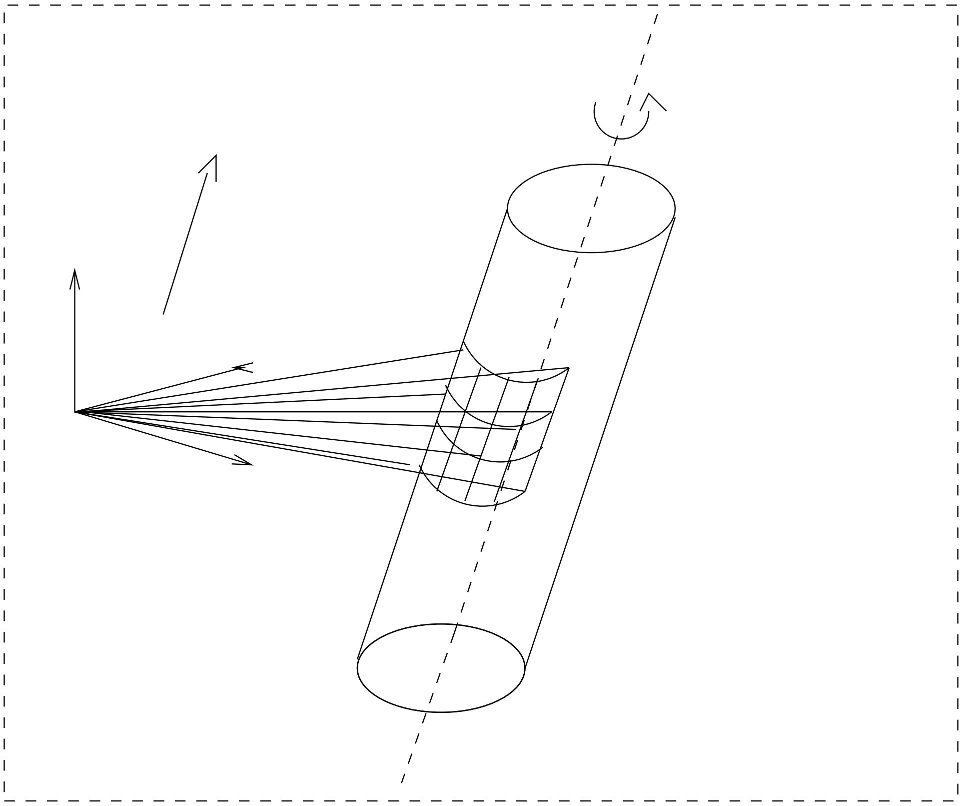
It is called a Point/Cylinder linkage, its class is 5.



In the case of a scanning telemeter the two translations orthogonal to the generating axis of the cylinder are constrained and the free motions are:

- the translation parallel to the axis of the cylinder
- the two rotations around the axes orthogonal to the axis of the cylinder passing by S
- the rotation around the axis of the cylinder

This linkage of class 4 is called Line/Cylinder linkage.



In the case of a range finder the only free motions are the translation parallel to the generating axis of the cylinder and the rotation around this axis. It is called a Prismatic/Revolute linkage.

4.2 Elaboration of sensor-based tasks

4.2.1 Sensor-based control

The aim of this paragraph is to explain the applications of these computations and the role of the matrix L^T in these applications. We introduce the sensor-based control defined by B. Espiau ([ECR90]) using the task function approach ([SEB90]) as it was used in [Cha90, CRE93]. The reader is referred to [SEB90] for more details.

Within this formalism, a robot task can be expressed by a $\mathcal{C}^2$ task function $e(\underline{r}, t)$ which must vanish over an interval of time $[0, T]$.

The problem is well-posed if:

- there exists an unique ideal trajectory $q_r(t)$ such that

$$e\,(q_r(t), t\,) = 0 \forall t \in [0, T] \text{ and } q_r(0) = q_0.$$

- the task Jacobian $\frac{\partial e}{\partial q}(q_r, t)$ is invertible.

The task function is then called *admissible*.

In the case of sensor-based control the task function will be elaborated directly from the sensor signals:

$$e_1 = C(\underline{s}\,(\underline{r}, t) \; - s_d(t))$$

where s_d is the desired sensor signal at time t and C is a full rank combination matrix, defined by the user and such that $C.L^T$ is a full rank matrix along $q_r(t)$.

The Jacobian matrix associated with the sensor-based task is :

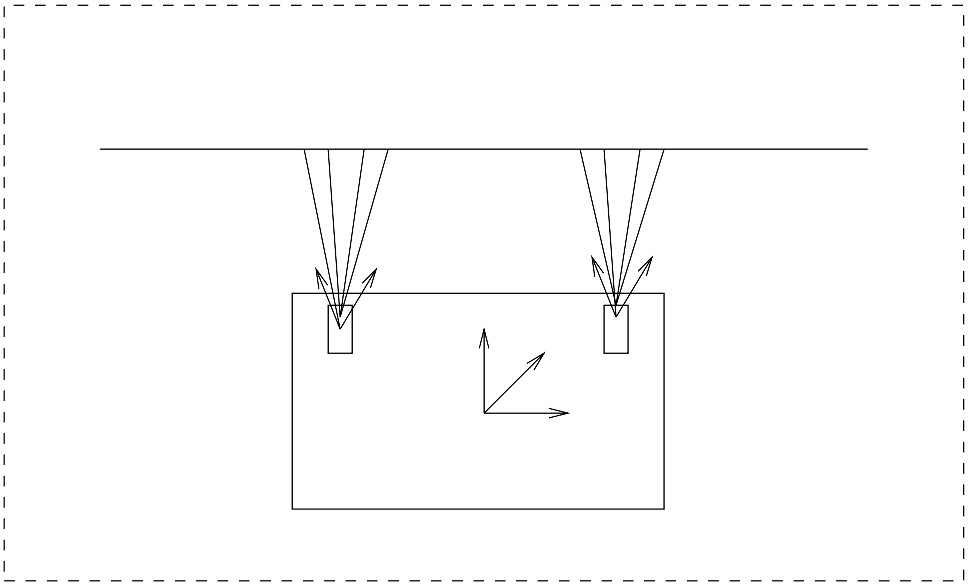
$$\frac{\partial e_1}{\partial \underline{r}} = C.\frac{\partial\,(\underline{s}\,(\underline{r}, t) - s_d(t)\,)}{\partial \underline{r}} = C.L^T$$

This will induce the choice of the sensor signals we will use to specify a given robot task. The number and the type of sensor signals must be so that L^T is a full rank matrix. See [Cha90, PG93] for more details about the specification of vision-based tasks and tasks connections for a control point of view. Here we are simply interested by finding a way of planning those tasks transitions, from the geometric interpretation of the sensor-based tasks.

4.2.2 Combining several elementary sensors

As we saw in section 3.1.2., the number of degrees of freedom allowed by a sensor-based task for one range sensor is often too big to specify a task only from this sensor signal. A way to solve this problem is simply to take into account *at the same time* signals from different sensors. This will introduce some new constraints and then lower the number of degrees of freedom of the robot.

For instance, let us consider the case of a wall following at a fixed distance d_r, with scanside sonars linked to a robot with a planar motion. For a single sounder, only the translation in the plane parallel to the wall, and the rotation around the normal axis to the plane passing by the point S are left free. An efficient mean to eliminate this rotation is to add a second sounder on the same side of the robot, and consider simultaneously the value d_2 of its signal with d_1, the value of the first sounder.



A classical control scheme is to express the conditions for translating into velocity constraints the geometric model for which $d_1 - d_2 = 0$ and $d_1 + d_2 - 2\,d_r = 0$ ([AA93]). A simpler mean, more robust because avoiding geometric modelling, is to impose d_r as sensing value for both sonars. This will be done by considering in the sensor-based task function the vector signal $\underline{s}(q,t) - \underline{s}_d(q,t)$ with $\underline{s}_d(q,t) = (d_r, d_r)$, while $\underline{s}_i(q,t)$ is the value given by the sensor i. To compute the free motions we will just have to express the interaction screws for each sensor in the the reference frame F_C of the robot thanks to the properties of the screws, and then stack the matrices ${L_i}^T$ obtained from these new screws.

Indeed if $(H(S), u)$ is the interaction screw computed in the frame F_S and if the frame F_C is obtained from F_S by a rotation R and a translation t, the interaction screw expressed in the frame F_C is $(R.(H(S) + u \times t), R.u)$.

This generic framework based on the concept of interaction screw can be applied for a range sensor, for a vision sensor, but also for several sensors of different nature provided that the constraints introduced by each of the sensors are compatible.

For instance, we can combine different data from sensors of different types into an interaction screw expressed in the control frame of the robot, in order to make more elaborate tasks (like positioning with respect to an object using a vision sensor and keeping the distance constant using a range sensor, whereas the vision sensor does not provide depth information).

Furthermore, we can combine several of these elementary models to form more elaborate structures, called sensor-based primitives.The design of each primitive is guided by the objective of simply achieving a certain virtual linkage characterized by a given interaction screw-subspace S.

The reader will find in appendix a catalog of standard sensor-based primitives representative of commonly encountered sensor-based tasks.

Chapter 5

Sensor-Based Primitives and Task Geometry

In the previous chapter generic models of the interaction between the sensor and its local environment were introduced. We have seen how they allow to specify the sensor-based task at hand in terms of constraints expressed in the sensor space, and to define a sensor-based task function (proportional to the distance between the current configuration and the desired configuration) to be minimized within a robust closed loop control scheme.

It emerges from this analysis that a natural way to specify a local sensor-based task is to define it in terms of geometrical constraints between the sensor frame and a frame linked to a particular part of the environment.

5.1 Analogy with Compliant Motion Planning

Keeping on with the analogy of kinematic of contacts, we will assimilate sensing constrained motions with compliant motions in the case of a manipulator.

Compliant motion occurs when the position of a manipulator is constrained by the task at hand. We are interested with active compliance implemented in the control servo loop, in order to define compliant motion primitives. The aim of compliant motion planning is to take tasks specifications expressed as operations on physical objects and derive an appropriate manipulator program using control system primitives.

Let us recall that the differential manifold of $\mathbb{R}^6$ separating the Configuration Space between a $\mathcal{C}$-obstacle and the Free Space is denoted a $\mathcal{C}$-surface ([Lat90]). For compliant motions a $\mathcal{C}$-surface can be viewed as a task configuration which allows only partial positioning freedom. We thus can consider a task configuration in terms of the resulting constraints on motion of the manipulator end-effector. Then, we can transform the constraints into a corresponding real-world control strategy ([Mas81]).

The task configuration is represented by a smooth hyper surface consisting of the possible positions of the ideal effector, called the *ideal* $\mathcal{C}$-surface. It is assumed to be connected and smooth, and its dimension, characterizing the number of Degrees Of Freedom (D. O. F.) allowed by the task, may vary from 0 to 6, like the one of its tangent space.

Since the ideal effector position lies in the ideal $\mathcal{C}$-surface at all times, the ideal effector velocity will lie in the vector subspace parallel to the tangent space. The goal trajectory is then the desired effector position as a function of time, which must lie in the ideal $\mathcal{C}$-surface.

By analogy, we can say that when the geometry linking two frames (the one of the sensor and the one of the 3D feature) constraints their relative positions, we can consider that we are in the case of a compliant motion.

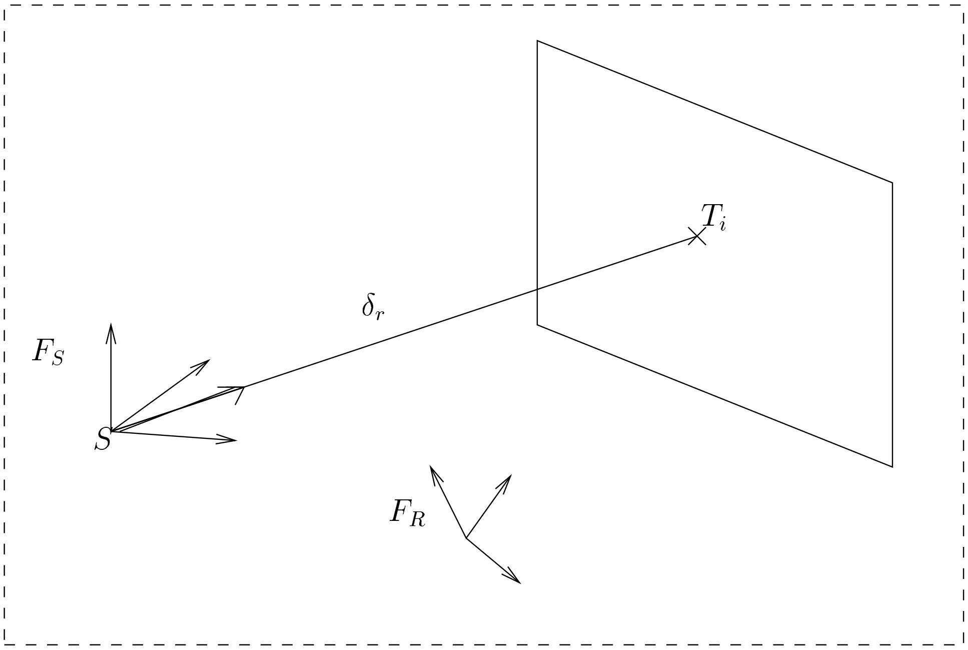
Our purpose is then to determine the "virtual" $\mathcal{C}$-surfaces and to produce a trajectory on them. The whole trajectory is a sequence composed of motions on "virtual" $\mathcal{C}$-surfaces and of motions linking two different $\mathcal{C}$-surfaces. Then we want to translate these artificial constraints into real-world control strategies.

5.2 Computation of the "virtual" $\mathcal{C}$-surfaces

As pointed out by Mason about compliant motion planning, the use of sub-manifolds of $\mathbb{R}^6$ to represent task constraints focuses directly on the most important characteristic of a compliant motion task (here a sensor-based task): the D. O. F. allowed during the task.

Let us now compute those "virtual" $\mathcal{C}$-surfaces :

For an elementary pin-point range sensor we will have the single constraint $ST_i = \delta_r$ where δ_r is the desired sensor value.



We express this in the fixed reference frame of the 3D scene F_R .

If (R, t) is the transformation from F_R to F_S, we have for S, $\vec{n_{S_i}}$ and T_i in the frame F_R:

$$\{ \ [S]_R = t [\vec{n_{S_i}}]_R = R.\vec{n_{S_i}} [T_i]_R = t + \delta_r R.n_{S_i} \tag{5.1}$$

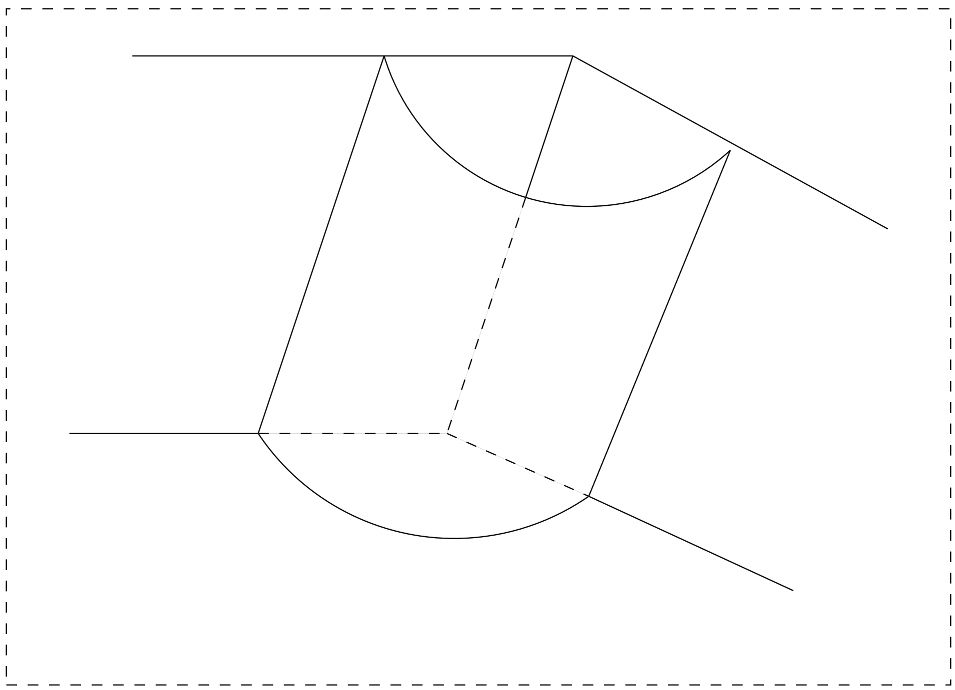
So we have an implicit equation of the desired surface in the reference frame of the 3D scene:

$$\vec{ST_i} = \delta_r . \vec{n_{S_i}} \tag{5.2}$$

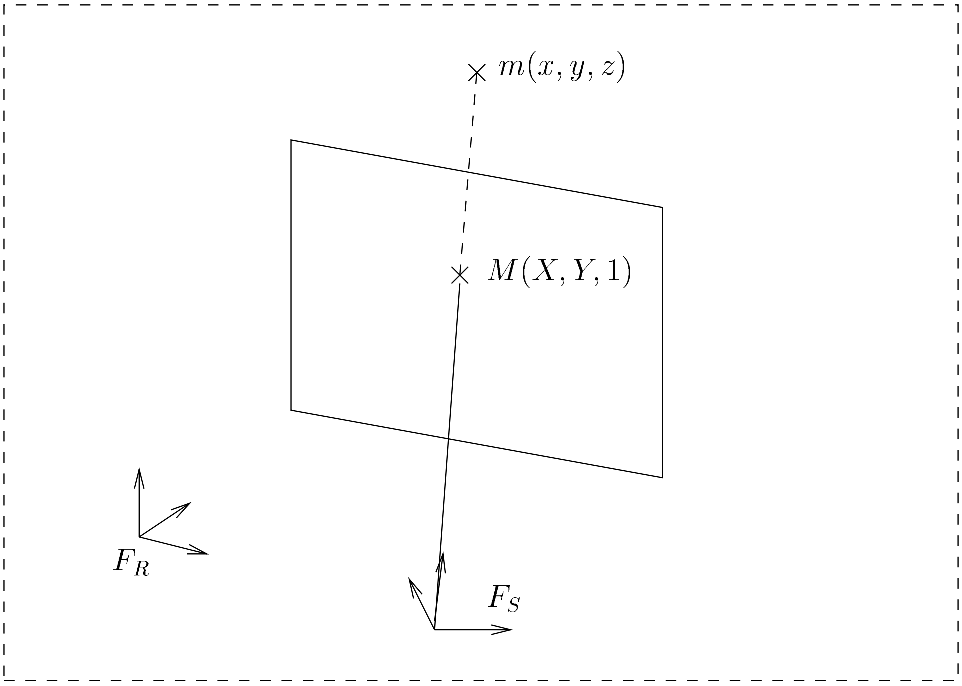
The dimension of this surface is 5, note that it is really equal to the class of the the virtual linkage (except in the special case of the edge, because of the discontinuity of the feature).

Consider now a scanside sonar taking two constraints given by two elementary pin-point sensors in a same plane,as for a 3D camera three non coplanar elementary pin-point sensors, we obtain easily a set of implicit equations depending on the geometric nature of the 3D feature considered.

For instance, the free surface for a scanside sonar or a 3D camera detecting an edge at a given distance d_r is:



For a vision sensor we will treat here only the case of a 3D point. The way to compute the surfaces for other 3D features by using the 2D features to observe ([Cha90]) is very similar. See Appendix A to see which are the quantities to be kept constant.



We have, in the frame F_S,

$$\begin{cases} X = \dfrac{x}{z} \\ \\ Y = \dfrac{y}{z} \end{cases}$$

The coordinates (x, y, z) of M are obtained by the change of frame formula $[M]_S = R.[M]_R + t$. This also gives us a set of implicit equations of a four-dimensional sub-manifold.

Chapter 6

Conclusion and future work

Conclusion

In order to cope with the problem of sensor-based navigation in an a priori known environment,for which we must build a robust control loop scheme, we have worked within the sensor-based control framework and pointed out a link between the sensor-based tasks specification it allows and motion planning.

We have defined sensor-based primitives for range sensors and have implemented a program for the generic computation of the interaction matrices to be used in the task function determination within control synthesis.

We have showed the geometric constraints arising from the sensor-based task between the robot and its local environment, and we have defined surfaces constrained by the desired values of the sensor signals.

The way of using the definition of these surfaces to transform the sequence *off-line trajectory planning/elaboration of a robust control scheme* in a single problem of sensor-based task planning has been posed.

Future work :

We will try to link to each sensor a validity domain (determined by visibility conditions) and free surfaces, parameterized by the desired sensor values in order to know which linkages can be realized in each part of the configuration space.

Once the free space is computed, we will take into account the robot own constraints for generating trajectories expressed directly in terms of sensor-servo controls. A trajectory will be expressed as a sequence of sensor-based tasks to achieve. A criterion to select a trajectory could be to minimize the number of different tasks in the sequence.

Catalog of canonic linkages for different types of sensors

Appendix A

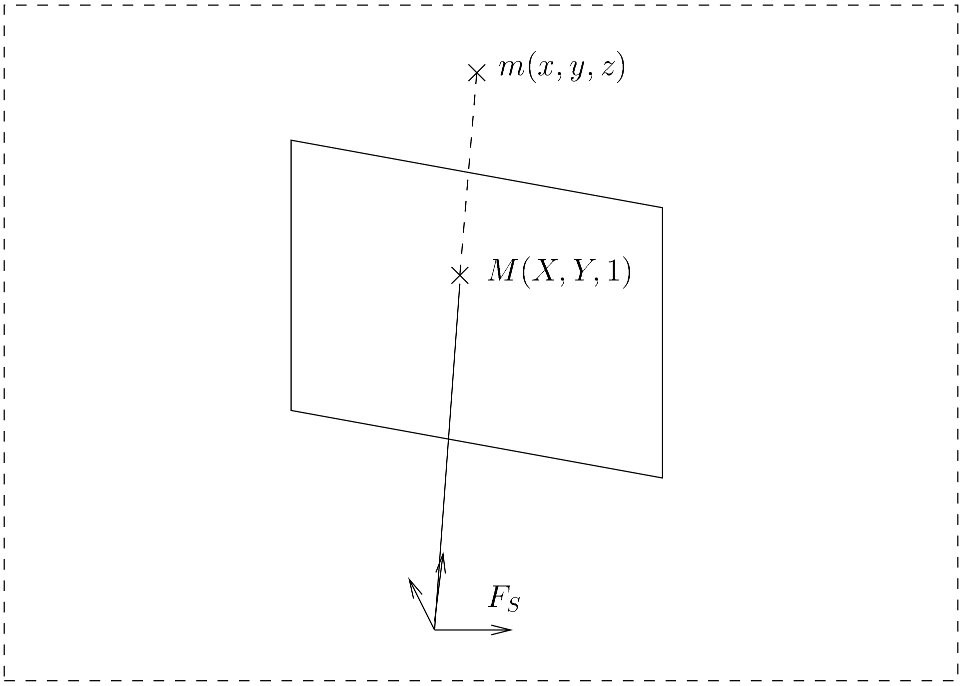
Vision Sensors

Computing the interaction matrix for vision sensors

Assuming a pinhole model of the camera, we defined ([Cha90]) a general rule to compute interaction matrices for the set of low level 3D geometrical primitives defined by a parametric function such that :

- point
- line
- plane
- quadric
- and so on ....

Each interaction matrix relates the variation of the sensor signal to the relative motion between the sensor and the 3D primitive. As example, the simplest interaction matrix relates the variation of the image coordinates $[X, Y]$ to the motion between the camera and a 3D point $[x, y, z]$ in the scene.



Such an interaction matrix can be easily derived by using the pinhole model of the camera and the general equations of the kinematics :

<u>**Pinhole model of the camera**</u>

$$\underline{X_i} = \frac{1}{z_i}.\underline{x_i}$$

with $\underline{X} = [X, Y, 1]^T$ and $\underline{x} = [x, y, z]^T$

Equations of the kinematic

$$\vec{V}(m) = \vec{V}(O) + \vec{\Omega} \times \vec{Om}$$

$$\Rightarrow \begin{cases} \dot{x} = V_x + \Omega_y z - \Omega_z y \\ \dot{y} = V_y + \Omega_z x - \Omega_x z \\ \dot{z} = V_z + \Omega_x y - \Omega_y x \end{cases}$$

Thus:

$$\begin{aligned} \dot{X} &= \frac{\dot{x}z - x\dot{z}}{z^2} \\ \dot{Y} &= \frac{\dot{y}z - y\dot{z}}{z^2} \end{aligned}$$

Interaction Matrix

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} -1/z & 0 & X/z & XY & -(1+X^2) & Y \\ 0 & -1/z & Y/z & 1+Y^2 & -XY & -X \end{bmatrix} \cdot \begin{bmatrix} V_x \\ V_y \\ V_z \\ \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix}$$

Note that we find here the well-known optical flow equation, which is a particular example of interaction matrix.

A.0.1 General computation rule:

We extended the previous result to the set of low level 3D features. This extension deals with the geometrical features in the scene which can be described by a set of parametric functions $h_i(\underline{x}, \underline{p}) = 0$. As example, a 3D plane will be expressed in the camera frame by :

$$h(\underline{x}, \underline{p}) = a_1 x + b_1 y + c_1 z + d_1 = 0$$

with $\underline{p} = [a_1, b_1, c_1, d_1] \in \Phi$

It is important to remark that $\underline{p}$ constitutes a chart of the configuration space Φ which characterizes all the possible attitudes (orientation and translation) of the 3D primitives in the camera frame.

Now, we state two propositions :

1. Assuming a pinhole model of the camera, then the 3D primitive projects onto the image frame as a 2D primitive which can be described by a parametric equation $g(\underline{X}, \underline{P}) = 0$ where $\underline{P}$ constitutes a chart of the configuration space Ψ which characterizes all the possible situations of the 2D primitives in the image frame.

2. At a given configuration $\underline{p_i}$ of the 3D primitive in the scene, corresponds an **unique** configuration $\underline{P_i}$ in the image (the converse is not true).

Indeed:

$$\begin{aligned} & g(\underline{X}, \underline{P}) = 0, \quad \forall \underline{X} \in \mathcal{P}_i \\ \Rightarrow \quad & \dot{g}(\underline{X}, \underline{P}) = 0, \quad \forall \underline{X} \in \mathcal{P}_i \end{aligned}$$

$$\frac{\partial g}{\partial \underline{P}}(\underline{X}, \underline{P})\ \underline{\dot{P}} + \frac{\partial g}{\partial \underline{X}}(\underline{X}, \underline{P})\ \underline{\dot{X}} = 0\ ,\ \forall \underline{X} \in \mathcal{P}_i$$

Substituting $\underline{\dot{X}}$ by the previously established optical flow equations $\underline{\dot{X}} = L_{of}^T\ T$ where $T = (V, \Omega)$ is the kinematic screw :

$$\frac{\partial g}{\partial \underline{P}}(\underline{X}, \underline{P})\ \underline{\dot{P}} + \frac{\partial g}{\partial \underline{X}}(\underline{X}, \underline{P})\ L_{of}^T\ T = 0\ ,\ \forall \underline{X} \in \mathcal{P}_i$$

Finally, we can express the change on the 2D primitive parameters observed in the image frame with regard to the relative motion between the camera and the 3D primitive in the scene:

$$\underline{\dot{P}} = -\left[\frac{\partial g}{\partial \underline{P}}\right]^{-1} \left[\frac{\partial g}{\partial \underline{X}}.L_{of}^T\right]\ T = L^T(\underline{p}, \underline{P})\ T$$

This general rule allows us to compute a closed form of the interaction matrix for the set of 3D geometrical primitives which projects onto the image frame under the form of a general parametric curve. As a consequence, we can use the parameters $\mathcal{P}_i$ which characterize the 2D features in the image frame as sensor signals [Cha90]. Hereafter, we give a list of interaction matrices corresponding to canonical geometrical primitives like line, plane, sphere...

Type of the geometric feature: 3 D points

Elementary screw:

$\underline{x} = (x, y, z)^T$

$\underline{X} = (X, Y, 1)^T$ with $\underline{X} = \frac{1}{z}\,\underline{x}$

$$L^T(\underline{X}, z) = \begin{pmatrix} -1/z & 0 & X/z & XY & -(1+X^2) & Y \\ 0 & -1/z & Y/z & 1+Y^2 & -XY & -X \end{pmatrix}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	2 D pattern	Sensor signals	Free space basis
Rigid	0	0	3 points	(x_i, y_i, z_i) for i=1..3	3 points	(X_i, Y_i) for i=1..3	$\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$
Prismatic	1	0					
Revolute	0	1					
Prismatic/ Revolute	1	1					
Plane/Plane Contact	2	1					
Ball-and-Socket	0	3					
Line/Plane Contact	2	2					
Line/Sphere Contact	1	3					
Ponctual Contact	2	3					

Type of the geometric feature: Straight lines

Elementary screw

A straight line is here represented as the intersection of two planes under the following form:

$$h(\underline{x}, \underline{p}) = \left\{ \begin{array}{l} a_1 x + b_1 y + c_1 z = 0 \\ a_2 x + b_2 y + c_2 z + d_2 = 0 \end{array} \right. \tag{A.1}$$

with $d_2 \neq 0$ in order to exclude degenerated cases. The equation of the projected line in the image plane is written:

$$g(\underline{X}, \underline{P}) = X \cos\theta + Y \sin\theta - \rho = 0 \text{ with } \left\{ \begin{array}{l} \cos\theta = a_1 / \sqrt{a_1^2 + b_1^2} \\ \sin\theta = b_1 / \sqrt{a_1^2 + b_1^2} \\ \rho = -c_1 / \sqrt{a_1^2 + b_1^2} \end{array} \right. \tag{A.2}$$

and we can obtain:

$$\begin{array}{l} L_\theta^T = [\begin{array}{cccccc} \lambda_\theta \cos\theta & \lambda_\theta \sin\theta & - - \lambda_\theta \rho & -\rho\cos\theta & -\rho\sin\theta & -1 \end{array}] \\ L_\rho^T = [\begin{array}{cccccc} \lambda_\rho \cos\theta & \lambda_\rho \sin\theta & - - \lambda_\rho \rho & (1+\rho^2)\sin\theta & -(1+\rho^2)\cos\theta & 0 \end{array}] \end{array} \tag{A.3}$$

with $\lambda_\theta = (a_2 \sin\theta - b_2 \cos\theta)/d_2$ and $\lambda_\rho = (a_2 \rho \cos\theta + b_2 \rho \sin\theta + c_2)/d_2$.

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	2 D pattern	Sensor signals	Free space basis
Rigid	0	0					
Prismatic	1	0	3 parallel lines	$\begin{cases} a_{1i}, b_{1i}, c_{1i}, d_{1i} \\ a_{2i}, b_{2i}, c_{2i}, d_{2i} \end{cases}$ for $i{=}1..3$	3 lines	(ρ_i, θ_i) for $i{=}1..3$	$\begin{pmatrix} a \\ b \\ c \\ 0 \\ 0 \\ 0 \end{pmatrix}$
			2 concurrent lines in (x^*, y^*, z^*)	(x^*, y^*, z^*), $\begin{cases} a_{1i}, b_{1i}, c_{1i}, d_{1i} \\ a_{2i}, b_{2i}, c_{2i}, d_{2i} \end{cases}$ for $i{=}1,2$	2 concurrent lines in (X^*, Y^*)	(X^*, Y^*), (ρ_i, θ_i) for $i{=}1,2$	$\begin{pmatrix} X^* \\ Y^* \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$
			3 concurrent lines in (x^*, y^*, z^*)	$\begin{cases} a_{1i}, b_{1i}, c_{1i}, d_{1i} \\ a_{2i}, b_{2i}, c_{2i}, d_{2i} \end{cases}$ for $i{=}1..3$	3 concurrent lines in (X^*, Y^*)	(ρ_i, θ_i) for $i{=}1..3$	$\begin{pmatrix} X^* \\ Y^* \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$
Revolute	0	1					
Prismatic/ Revolute	1	1					
Plane/Plane Contact	2	1					
Ball-and-Socket	0	3					
Line/Plane Contact	2	2					
Line/Sphere Contact	1	3					
Ponctual Contact	2	3					

Type of the geometric feature: 3 D segments

Elementary screw:

- Segment represented by its ends $(X_1, Y_1), (X_2, Y_2)$:

$$L^T = \begin{pmatrix} -1/z_1 & 0 & X_1/z_1 & X_1\,Y_1 & -(1+X_1^2) & Y_1 \\ 0 & -1/z_1 & Y_1/z_1 & 1+Y_1^2 & -X_1\,Y_1 & -X_1 \\ -1/z_2 & 0 & X_2/z_2 & X_2\,Y_2 & -(1+X_2^2) & Y_2 \\ 0 & -1/z_2 & Y_2/z_2 & 1+Y_2^2 & -X_2\,Y_2 & -X_2 \end{pmatrix}$$

- Segment represented by its center (X_c, Y_c), its length l and its orientation α:

$$\begin{array}{lccl}
L^T{}_{X_c} = (& -\lambda_2 & 0 & \\
 & \lambda_2\,X_c - \lambda_1\,l\,\cos\alpha/4 & X_c\,Y_c + l^2\,\cos\alpha\,\sin\alpha/4 & \\
 & -(1+X_c^2+l^2\,\cos^2\alpha/4) & Y_c &) \\
L^T{}_{Y_c} = (& 0 & -\lambda_2 & \\
 & \lambda_2\,Y_c - \lambda_1\,l\,\sin\alpha/4 & 1+Y_c^2+l^2\,\sin^2\alpha/4 & \\
 & -X_c\,Y_c - l^2\,\cos\alpha\sin\alpha/4 & -X_c &) \\
L^T{}_{l} = (& \lambda_1\,\cos\alpha & \lambda_1\,\sin\alpha & \\
 & \lambda_2\,l - \lambda_1(X_c\,\cos\alpha + Y_c\,\sin\alpha) & l[X_c\,\cos\alpha\,\sin\alpha + Y_c(1+\sin^2\alpha)] & \\
 & -l[X_c(1+\cos^2\alpha) + Y_c\,\cos\alpha\,\sin\alpha] & 0 &) \\
L^T{}_{\alpha} = (& -\lambda_1\,\sin\alpha/l & \lambda_1\,\cos\alpha/l & \\
 & \lambda_1(X_c\,\sin\alpha - Y_c\,\cos\alpha)/l & -X_c\,\sin^2\alpha + Y_c\,\cos\alpha\sin\alpha & \\
 & X_c\,\cos\alpha\,\sin\alpha - Y_c\,\cos^2\alpha & -1 &)
\end{array}$$

avec $\lambda_1 = \dfrac{z_1 - z_2}{z_1 z_2}$ et $\lambda_2 = \dfrac{z_1 + z_2}{2 z_1 z_2}$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	2 D pattern	Sensor signals	Free space basis
Rigid	0	0					
Prismatic	1	0	2 segments concurrents in their center in $(0, 0, z^*)$	(x_{1i}, y_{1i}, z^*) (x_{2i}, y_{2i}, z^*) for $i{=}1..2$	2 segments concurrents in their center in $(0, 0)$	$z^*, (l_i, \alpha_i)$ for $i{=}1..2$	$\begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$
Revolute	0	1	2 segments concurrents in their center in $(0, 0, z^*)$	(x_i, y_i, z_i) for $i{=}1{,}2$	2 segments concurrents in their center in $(0, 0)$	$z^*, (l_i, \alpha_i)$ for $i{=}1{,}2$	$\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$
Prismatic/ Revolute	1	1					
Plane/Plane Contact	2	1	3 segments at least of a regular networking parallel to the image plane at a distance z^*	$z^*, (l_i)$ for $i = 1..3$	3 segments of a regular networking	z^*, (X_{c_i}, Y_{c_i}), (λ_i, α_i) for $i{=}1..3$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$
Ball-and-Socket	0	3					
Line/Plane Contact	2	2					
Line/Sphere Contact	1	3					
Ponctual Contact	2	3					

Type of the geometric feature: Circle

Elementary screw:

A circle may be represented as the intersection of a sphere and a plane:

$$h(\underline{x},\underline{p}) = \begin{cases} (x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2 - r^2 = 0 \\ \alpha(x-x_0) + \beta(y-y_0) + \gamma(z-z_0) = 0 \end{cases} \tag{A.4}$$

Its projection on the image plane has the form of an ellipse or a circle (except in degenerated cases where the projection is a segment), which may be represented as:

$$g(\underline{X},\underline{P}) = \frac{(X - X_c + E(Y-Y_c))^2}{A^2(1+E^2)} + \frac{(Y - Y_c - E(X-X_c))^2}{B^2(1+E^2)} - 1 = 0 \tag{A.5}$$

$$\text{with} \begin{cases} X_c = (A_1A_3 - A_2A_4)/(A_2^2 - A_1) \\ Y_c = (A_0A_4 - A_2A_3)/(A_2^2 - A_1) \\ E = (A_1 - 1 \pm \sqrt{(A_1-1)^2 + 4A_2^2})/2A_2 \\ A^2 = 2(X_c^2 + 2A_2X_cY_c + A_1Y_c^2 - A_5)/(1 + A_1 \pm \sqrt{(A_1-1)^2 + 4A_2^2}) \\ B^2 = 2(X_c^2 + 2A_2X_cY_c + A_1Y_c^2 - A_5)/(1 + A_1 \mp \sqrt{(A_1-1)^2 + 4A_2^2}) \end{cases}$$

where the coefficients A_i's are obtained from the polynomial equation of an ellipse:

$$g(\underline{X},\underline{P}) \ : X^2 + A_1Y^2 + 2A_2XY + 2A_3X + 2A_4Y + A_5 = 0 \tag{A.6}$$

More precisely, we have:

$$\begin{cases} A_1 = [b^2(x_0^2 + y_0^2 + z_0^2 - r^2) + 1 - 2by_0]/A_0 \\ A_2 = [ab(x_0^2 + y_0^2 + z_0^2 - r^2) - bx_0 - ay_0]/A_0 \\ A_3 = [ac(x_0^2 + y_0^2 + z_0^2 - r^2) - cx_0 - az_0]/A_0 \\ A_4 = [bc(x_0^2 + y_0^2 + z_0^2 - r^2) - cy_0 - bz_0]/A_0 \\ A_5 = [c^2(x_0^2 + y_0^2 + z_0^2 - r^2) + 1 - 2cz_0]/A_0 \end{cases} \tag{A.7}$$

whith $A_0 = a^2(x_0^2 + y_0^2 + z_0^2 - r^2) + 1 - 2ax_0 \quad (\neq 0)$
and $a = \alpha/(\alpha x_0 + \beta y_0 + \gamma z_0)$, $b = \beta/(\alpha x_0 + \beta y_0 + \gamma z_0)$, $c = \gamma/(\alpha x_0 + \beta y_0 + \gamma z_0)$.

The natural parametrization (X_c, Y_c, E, A, B) is always ambiguous. Furthermore, the orientation E is not defined when the ellipse is a circle. So we use a representation based on the normalized inertial moments $(X_c, Y_c, \mu_{20}, \mu_{11}, \mu_{02})$ with:

$$\begin{cases} \mu_{20} = (A^2 + B^2E^2)/(1+E^2) \\ \mu_{11} = E(A^2 - B^2)/(1+E^2) \\ \mu_{02} = (A^2E^2 + B^2)/(1+E^2) \end{cases} \tag{A.8}$$

and we can obtain:

$$
\begin{array}{llll}
L^T_{X_c} = [& -1/z_c & 0 & X_c/z_c + a\mu_{20} + b\mu_{11} \\
 & X_cY_c + \mu_{11} & -1 - X_c^2 - \mu_{20} & Y_c \quad] \\
L^T_{Y_c} = [& 0 & -1/z_c & Y_c/z_c + a\mu_{11} + b\mu_{02} \\
 & 1 + Y_c^2 + \mu_{02} & -X_cY_c - \mu_{11} & -X_c \quad] \\
L^T_{\mu_{20}} = [& -2(a\mu_{20} + b\mu_{11}) & 0 & 2[(1/z_c + aX_c)\mu_{20} + bX_c\mu_{11}] \\
 & 2(Y_c\mu_{20} + X_c\mu_{11}) & -4\mu_{20}X_c & 2\mu_{11} \quad] \\
L^T_{\mu_{11}} = [& -a\mu_{11} - b\mu_{02} & -a\mu_{20} - b\mu_{11} & aY_c\mu_{20} + (3/z_c - c)\mu_{11} + bX_c\mu_{02} \\
 & 3Y_c\mu_{11} + X_c\mu_{02} & -Y_c\mu_{20} - 3X_c\mu_{11} & \mu_{02} - \mu_{20} \quad] \\
L^T_{\mu_{02}} = [& 0 & -2(a\mu_{11} + b\mu_{02}) & 2[(1/z_c + bY_c)\mu_{02} + aY_c\mu_{11}] \\
 & 4Y_c\mu_{02} & -2(Y_c\mu_{11} + X_c\mu_{02}) & -2\mu_{11} \quad]
\end{array}
\tag{A.9}
$$

with $z_c = 1/(aX_c + bY_c + c)$.

Virtual Linkage	Class		3 D pattern	3 D parameters	2 D pattern	Sensor signals	Free space basis
	T	R					
Rigid	0	0					
Prismatic	1	0					
Revolute	0	1	2 circles in a same fronto plane centered in $(0,0,z^*)$	$\begin{cases}(0,0,z_i),r_i \\ \text{for } i{=}1..2 \\ \alpha,\beta,\gamma\end{cases}$	2 circles centered in $(0,0)$	R_1, R_2, z^*	$\begin{pmatrix}0\\0\\0\\0\\0\\1\end{pmatrix}$
			2 circles in two different fronto planes of same radius r centered in $(0,0,z^*)$	$\begin{cases}(0,0,z^*),r \\ \alpha_i,\beta_i,\gamma_i \\ \text{for } i{=}1..2\end{cases}$	2 circles centered in $(0,0)$	R_1, R_2, z^*	$\begin{pmatrix}0\\0\\0\\0\\0\\1\end{pmatrix}$
			1 circle in a fronto plane centered in $(0,0,z)$ and a point $(0,0,z^*)$	$\begin{cases}(0,0,z),r \\ (0,0,z^*)\end{cases}$	1 circle of radius R centered in $(0,0)$	R, z^*	$\begin{pmatrix}0\\0\\0\\0\\0\\1\end{pmatrix}$
Prismatic/ Revolute	1	1					
Plane/Plane Contact	2	1					
Ball-and- Socket	0	3					
Line/Plane Contact	2	2					
Line/Sphere Contact	1	3					
Ponctual Contact	2	3					

Type of the geometric feature: Cylinder

Elementary screw:

A cylinder may be represented by the equation:

$$h(\underline{x},\underline{p}) = (x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2 - -(ax+by+cz)^2 - r^2 = 0 \tag{A.10}$$

$$\text{with} \begin{cases} a^2+b^2+c^2 = 1 \\ ax_0+by_0+cz_0 = 0 \end{cases}$$

where r is the radius of the cylinder, a, b and c are the coordinates of its direction vector and x_0, y_0 and z_0 are the coordinates of the nearest point belonging to the cylinder axis from the projection center.

The projection of a cylinder on the image plane is (for non-degenerated cases) a set of two straight lines with equation:

$$\begin{cases} D_1 & : & X\cos\theta_1 + Y\sin\theta_1 - \rho_1 = 0 \\ D_2 & : & X\cos\theta_2 + Y\sin\theta_2 - \rho_2 = 0 \end{cases} \tag{A.11}$$

with:

$$\begin{cases} \cos\theta_1 = \dfrac{rx_0/A-\alpha}{\sqrt{(rx_0/A-\alpha)^2+(ry_0/A-\beta)^2}}\ , \cos\theta_2 = \dfrac{rx_0/A+\alpha}{\sqrt{(rx_0/A+\alpha)^2+(ry_0/A+\beta)^2}} \\ \sin\theta_1 = \dfrac{ry_0/A-\beta}{\sqrt{(rx_0/A-\alpha)^2+(ry_0/A-\beta)^2}}\ , \sin\theta_2 = \dfrac{ry_0/A+\beta}{\sqrt{(rx_0/A+\alpha)^2+(ry_0/A+\beta)^2}} \\ \rho_1 = \dfrac{rz_0/A-\gamma}{\sqrt{(rx_0/A-\alpha)^2+(ry_0/A-\beta)^2}}\ , \rho_2 = \dfrac{rz_0/A+\gamma}{\sqrt{(rx_0/A+\alpha)^2+(ry_0/A+\beta)^2}} \end{cases}$$

where :

$$\begin{cases} A = \sqrt{x_0^2+y_0^2+z_0^2-(ax_0+by_0+cz_0)^2-r^2} \\ \alpha = cy_0 - bz_0 \\ \beta = az_0 - cx_0 \\ \gamma = bx_0 - ay_0 \end{cases}$$

and we finally obtain:

$$\begin{array}{llllllll} L_{\rho_1}^T & = [& \lambda_{\rho_1}\cos\theta_1 & \lambda_{\rho_1}\sin\theta_1 & -\,-\,\lambda_{\rho_1}\rho_1 & (1+\rho_1^2)\sin\theta_1 & -\,-\,(1+\rho_1^2)\cos\theta_1 & 0 &] \\ L_{\theta_1}^T & = [& \lambda_{\theta_1}\cos\theta_1 & \lambda_{\theta_1}\sin\theta_1 & -\lambda_{\theta_1}\rho_1 & -\,-\,\rho_1\cos\theta_1 & -\rho_1\sin\theta_1 & -1 &] \\ L_{\rho_2}^T & = [& \lambda_{\rho_2}\cos\theta_2 & \lambda_{\rho_2}\sin\theta_2 & -\,-\,\lambda_{\rho_2}\rho_2 & (1+\rho_2^2)\sin\theta_2 & -\,-\,(1+\rho_2^2)\cos\theta_2 & 0 &] \\ L_{\theta_2}^T & = [& \lambda_{\theta_2}\cos\theta_2 & \lambda_{\theta_2}\sin\theta_2 & -\,-\,\lambda_{\theta_2}\rho_2 & -\rho_2\cos\theta_2 & -\,-\,\rho_2\sin\theta_2 & -1 &] \end{array} \tag{A.12}$$

$$\text{with} \begin{cases} \lambda_{\rho_1} & = & -(x_0\rho_1\cos\theta_1 + y_0\rho_1\sin\theta_1 + z_0)/A^2 \\ \lambda_{\theta_1} & = & (y_0\cos\theta_1 - x_0\sin\theta_1)/A^2 \\ \lambda_{\rho_2} & = & -(x_0\rho_2\cos\theta_2 + y_0\rho_2\sin\theta_2 + z_0)/A^2 \\ \lambda_{\theta_2} & = & (y_0\cos\theta_2 - x_0\sin\theta_2)/A^2 \end{cases}$$

Now let us consider a cylinder with axis parallel to the image plane and passing by the point $(0, 0, z^*)$ Its projection on the image plane is made of two parallel lines symmetrical about the origin of the image plane which equations are :

$$\left\{ \begin{array}{ll} D_1 : & b\,X - a\,Y + R/\sqrt{z^{*2} - R^2} = 0 \\ D_2 : & -b\,X + a\,Y + R/\sqrt{z^{*2} - R^2} = 0 \end{array} \right.$$

Representing those lines with (ρ, θ) parametrization we obtain :

$$\begin{array}{lcccccccc} {L^T}_{\rho_1} = [& \lambda_\rho & -\lambda_\rho\,a & -\lambda_\rho\,\rho & -a(1+\rho^2) & -b(1+\rho^2) & 0 &] \\ {L^T}_{\theta_1} = [& 0 & 0 & 0 & -b\,\rho & -a\,\rho & -1 &] \\ {L^T}_{\rho_2} = [& -\lambda_\rho\,b & \lambda_\rho\,a & -\lambda_\rho\,\rho & a(1+\rho^2) & b(1+\rho^2) & 0 &] \\ {L^T}_{\theta_2} = [& 0 & 0 & 0 & b\,\rho & -a\,\rho & -1 &] \end{array}$$

with $\lambda_\rho = \dfrac{-z^*}{z^{*2} - R^2}$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	2 D pattern	Sensor signals	Free space basis
Rigid	0	0					
Prismatic	1	0					
Revolute	0	1					
Prismatic/ Revolute	1	1	1 cylinder of radius R with generating axis parallel to the image plane and passing by the point $(0,0,z^*)$ 1 cylinder of radius R with generating axis parallel to (Oz)	a, b, R, z^* R	2 parallel lines symmetrical about the origin of the image plane 2 circles or 2 concurrent segments	$(\lambda, \rho), a, b$ $\left\{ \begin{array}{l} (X_{c_i}, Y_{c_i}) \\ \text{for } i{=}1..2 \\ R_1, r_2, z^* \end{array} \right.$ $\left\{ \begin{array}{l} (X_{c_i}, Y_{c_i}) \\ (l_i, \alpha_i) \\ \text{for } i{=}1..2 \end{array} \right.$	$\begin{pmatrix} a & 0 \\ b & 0 \\ 0 & 0 \\ 0 & a \\ 0 & b \\ 0 & 0 \end{pmatrix}$ $\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}$
Plane/Plane Contact	2	1					
Ball-and-Socket	0	3					
Line/Plane Contact	2	2					
Line/Sphere Contact	1	3					
Ponctual Contact	2	3					

Type of the geometric feature: Sphere

Elementary screw:

A sphere is represented by:

$$h(\underline{x}, \underline{p}) = (x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 - r^2 = 0 \tag{A.13}$$

The image of a sphere is an ellipse (a circle when $x_0 = y_0 = 0$) with equation:

$$g(\underline{X}, \underline{P}) \ : X^2 + A_1 Y^2 + 2A_2 XY + 2A_3 X + 2A_4 Y + A_5 = 0 \tag{A.14}$$

$$\text{with} \left\{ \begin{array}{l} A_1 = (r^2 - x_0^2 - z_0^2)/(r^2 - y_0^2 - z_0^2) \\ A_2 = x_0 y_0/(r^2 - y_0^2 - z_0^2) \\ A_3 = x_0 z_0/(r^2 - y_0^2 - z_0^2) \\ A_4 = y_0 z_0/(r^2 - y_0^2 - z_0^2) \\ A_5 = (r^2 - x_0^2 - y_0^2)/(r^2 - y_0^2 - z_0^2) \end{array} \right.$$

The interaction matrix related to an ellipse has already been given. Results are therefore given by Eq. (A.7) with the particular following values for a, b and c:

$$\left\{ \begin{array}{l} a = x_0/(x_0^2 + y_0^2 + z_0^2 - r^2) \\ b = y_0/(x_0^2 + y_0^2 + z_0^2 - r^2) \\ c = z_0/(x_0^2 + y_0^2 + z_0^2 - r^2) \end{array} \right. \tag{A.15}$$

that describe the plane in which belong the limb points of the sphere through:

$$1/z = aX + bY + c \tag{A.16}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	2 D pattern	Sensor signals	Free space basis
Rigid	0	0					
Prismatic	1	0					
Revolute	0	1					
Prismatic/ Revolute	1	1					
Plane/Plane Contact	2	1					
Ball-and-Socket	0	3	1 sphere of radius R centered in $(0, 0, z^*)$	R, z^*	1 circle of radius R centered in the origin	$X_c, Y_c, \mu_{20}, \mu_{02}$	$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$
Line/Plane Contact	2	2					
Line/Sphere Contact	1	3					
Ponctual Contact	2	3					

Appendix B

Range Sensors

Type of the sensor: Punctual sensor

Type of the geometric feature: Edge

Elementary screw:

$$L^T = \begin{pmatrix} -1 & -\dfrac{b_1}{a_1} & -\dfrac{c_1}{a_1} & 0 & -\dfrac{\delta c_1}{a_1} & \dfrac{\delta b_1}{a_1} \\ -1 & -\dfrac{b_2}{a_2} & -\dfrac{c_2}{a_2} & 0 & -\dfrac{\delta c_2}{a_2} & \dfrac{\delta b_2}{a_2} \end{pmatrix}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0				
Revolute	0	1				
Prismatic/ Revolute	1	1				
Plane/plane contact	2	1				
Ball-and-socket	0	3				
Line/plane contact	2	2				
Line/sphere contact	1	3	intersection of 2 planes	$\left\{ \begin{array}{l} a_1, b_1, c_1, d_1 \\ a_2, b_2, c_2, d_2 \end{array} \right.$	δ	$\begin{pmatrix} c_1 b_2 - c_2 b_1 & \delta(b_1 c_2 - b_2 c_1) & 0 & 0 \\ a_1 c_2 - a_2 c_1 & \delta(a_2 c_1 - a_1 c_2) & \delta & 0 \\ a_2 b_1 - a_1 b_2 & 0 & 0 & 0 \\ a_2 b_1 - a_1 b_2 & 0 & 0 & 1 \\ 0 & a_1 b_2 - a_2 b_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$
Point/plane contact	2	3				

Type of the sensor: Punctual sensor

Type of the geometric feature: Plane

Elementary screw:

$$L^T = \begin{pmatrix} -1 & -\dfrac{b}{a} & -\dfrac{c}{a} & 0 & -\dfrac{\delta c}{a} & \dfrac{\delta b}{a} \end{pmatrix}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0				
Revolute	0	1				
Prismatic/ Revolute	1	1				
Plane/plane contact	2	1				
Ball-and-socket	0	3				
Line/plane contact	2	2				
Line/sphere contact	1	3				
Point/plane contact	2	3	a plane	(a, b, c, d)	δ	$\begin{pmatrix} -c & -b & -\delta c & \delta b & 0 \\ 0 & a & 0 & 0 & 0 \\ a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & a & 0 & 0 \\ 0 & 0 & 0 & a & 0 \end{pmatrix}$

Type of the sensor: Punctual sensor

Type of the geometric feature: Sphere

Elementary screw:

$$L^T = \begin{pmatrix} -1 & -\frac{y_0}{-s_1+x_0} & -\frac{z_0}{-s_1+x_0} & 0 & -\frac{\delta z_0}{-s_1+x_0} & \frac{\delta y_0}{-s_1+x_0} \end{pmatrix}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0				
Revolute	0	1				
Prismatic/ Revolute	1	1				
Plane/plane contact	2	1				
Ball-and-socket	0	3				
Line/plane contact	2	2				
Line/sphere contact	1	3				
Point/plane contact	2	3	a sphere	$(x_0, y_0, z_0), R$	δ	$\begin{pmatrix} 0 & y_0 & 0 & 0 & 0 \\ -z_0 & \delta - x_0 & \delta & 0 & -\delta\, z_0 \\ y_0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & y_0 & 0 & y_0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$

Type of the sensor: Punctual sensor

Type of the geometric feature: Cylinder

Elementary screw:

$$L = \begin{pmatrix} -1 \\ -\frac{y_0\, b^2+y_0+z_0\, bc-ab\delta+abx_0}{b^2\delta-b^2x_0-2\,\delta+2\,x_0+c^2\delta-c^2x_0+z_0\, ac+y_0\, ab} \\ -\frac{z_0+z_0\, c^2-ac\delta+acx_0+y_0\, bc}{b^2\delta-b^2x_0-2\,\delta+2\,x_0+c^2\delta-c^2x_0+z_0\, ac+y_0\, ab} \\ 0 \\ -\frac{\delta\left(z_0+z_0\, c^2-ac\delta+acx_0+y_0\, bc\right)}{b^2\delta-b^2x_0-2\,\delta+2\,x_0+c^2\delta-c^2x_0+z_0\, ac+y_0\, ab} \\ \frac{\delta\left(y_0\, b^2+y_0+z_0\, bc-ab\delta+abx_0\right)}{b^2\delta-b^2x_0-2\,\delta+2\,x_0+c^2\delta-c^2x_0+z_0\, ac+y_0\, ab} \end{pmatrix}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0				
Revolute	0	1				
Prismatic/ Revolute	1	1				
Plane/plane contact	2	1				
Ball-and-socket	0	3				
Line/plane contact	2	2				
Line/sphere contact	1	3				
Point/plane contact	2	3	a cylinder of radius R with axis $(a, b, c)^T$ passing by (x_0, y_0, z_0)	x_0, y_0, z_0 a, b, c R	δ	$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ s_1 & s_2 & 0 & s_3 & \delta \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$

with

$$\begin{aligned}
s_1 &= \frac{2\,\delta - 2\,x_0 + (-\delta + x_0)\,c^2 - z_0\,ac + (-\delta + x_0)\,b^2 - y_0\,ab}{y_0\,b^2 + y_0 + z_0\,bc + (-\delta + x_0)\,ab}\\
s_2 &= \frac{-y_0\,bc - z_0 - z_0\,c^2 + (\delta - x_0)\,ca}{y_0\,b^2 + y_0 + z_0\,bc + (-\delta + x_0)\,ab}\\
s_3 &= \frac{-\delta y_0\,bc - \delta z_0 - \delta z_0\,c^2 - \delta\,(-\delta + x_0)\,ca}{y_0\,b^2 + y_0 + z_0\,bc + (-\delta + x_0)\,ab}
\end{aligned}$$

Type of the sensor: Linear sensor

Type of the geometric feature: Edge

Elementary screw:

$$H_i = \begin{pmatrix} -\dfrac{a_1}{\cos\theta_i a_1 + \sin\theta_i b_1} & -\dfrac{a_2}{\cos\theta_i a_2 + \sin\theta_i b_2} \\ -\dfrac{b_1}{\cos\theta_i a_1 + \sin\theta_i b_1} & -\dfrac{b_2}{\cos\theta_i a_2 + \sin\theta_i b_2} \\ -\dfrac{c_1}{\cos\theta_i a_1 + \sin\theta_i b_1} & -\dfrac{c_2}{\cos\theta_i a_2 + \sin\theta_i b_2} \\ \dfrac{\delta_i c_1 \ \sin\theta_i}{\cos\theta_i a_1 + \sin\theta_i b_1} & \dfrac{\delta_i c_2 \ \sin\theta_i}{\cos\theta_i a_2 + \sin\theta_i b_2} \\ -\dfrac{\delta_i c_1 \ \cos\theta_i}{\cos\theta_i a_1 + \sin\theta_i b_1} & -\dfrac{\delta_i c_2 \ \cos\theta_i}{\cos\theta_i a_2 + \sin\theta_i b_2} \\ -\dfrac{\delta_i \left(a_1 \ \sin\theta_i - b_1 \ \cos\theta_i\right)}{\cos\theta_i a_1 + \sin\theta_i b_1} & -\dfrac{\delta_i \left(a_2 \ \sin\theta_i - b_2 \ \cos\theta_i\right)}{\cos\theta_i a_2 + \sin\theta_i b_2} \end{pmatrix}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0	the intersection of 2 planes	$\left\{ \begin{array}{l} a_1, b_1, c_1, d_1 \\ a_2, b_2, c_2, d_2 \end{array} \right.$	δ_i	$\begin{pmatrix} b_1\,c_2 - c_1\,b_2 \\ c_1\,a_2 - a_1\,c_2 \\ b_2\,a_1 - a_2\,b_1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$
Revolute	0	1				
Prismatic/ Revolute	1	1				
Plane/plane contact	2	1				
Ball-and-socket	0	3				
Line/plane contact	2	2				
Line/sphere contact	1	3				
Point/plane contact	2	3				

Type of the sensor: Linear sensor

Type of the geometric feature: Plane

Elementary screw:

$$H_i = \begin{pmatrix} -\dfrac{a}{\cos\theta_i a + \sin\theta_i b} \\ -\dfrac{b}{\cos\theta_i a + \sin\theta_i b} \\ -\dfrac{c}{\cos\theta_i a + \sin\theta_i b} \\ \dfrac{\delta_i c \sin\theta_i}{\cos\theta_i a + \sin\theta_i b} \\ -\dfrac{\delta_i c \cos\theta_i}{\cos\theta_i a + \sin\theta_i b} \\ \dfrac{\delta_i\left(-a\sin\theta_i + b\cos\theta_i\right)}{\cos\theta_i a + \sin\theta_i b} \end{pmatrix}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0				
Revolute	0	1				
Prismatic/ Revolute	1	1				
Plane/plane contact	2	1	a plane	(a, b, c, d)	δ_i	$\begin{pmatrix} -c & -b & 0 \\ 0 & a & 0 \\ a & 0 & 0 \\ 0 & 0 & a \\ 0 & 0 & b \\ 0 & 0 & c \end{pmatrix}$
Ball-and-socket	0	3				
Line/plane contact	2	2				
Line/sphere contact	1	3				
Point/plane contact	2	3				

Type of the sensor: Linear sensor

Type of the geometric feature: Sphere

Elementary screw:

$$H_i = \begin{pmatrix} -\dfrac{-\delta_i \cos\theta_i + x_0}{\cos\theta_i x_0 - \delta_i + \sin\theta_i y_0} \\ -\dfrac{-\delta_i \sin\theta_i + y_0}{\cos\theta_i x_0 - \delta_i + \sin\theta_i y_0} \\ -\dfrac{z_0}{\cos\theta_i x_0 - \delta_i + \sin(\theta_i) y_0} \\ \dfrac{\delta_i z_0 \ \sin\theta_i}{\cos\theta_i x_0 - \delta_i + \sin\theta_i y_0} \\ -\dfrac{\delta_i z_0 \ \cos\theta_i}{\cos\theta_i x_0 - \delta_i + \sin\theta_i y_0} \\ -\dfrac{\delta_i \left(x_0 \ \sin\theta_i - \cos\theta_i y_0\right)}{\cos\theta_i x_0 - \delta_i + \sin\theta_i y_0} \end{pmatrix}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0				
Revolute	0	1				
Prismatic/ Revolute	1	1				
Plane/plane contact	2	1				
Ball-and-socket	0	3	a sphere of radius R centered in (x_0, y_0, z_0)	x_0, y_0, z_0, R	δ_i	$\begin{pmatrix} 0 & -y_0 & z_0 \\ -z_0 & x_0 & 0 \\ y_0 & 0 & -x_0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$
Line/plane contact	2	2				
Line/sphere contact	1	3				
Point/plane contact	2	3				

Type of the sensor: Linear sensor

Type of the geometric feature: Cylinder

Elementary screw:

$$
H_i = \begin{pmatrix}
-\dfrac{2\,\delta_i \cos\theta_i - \delta_i \cos\theta_i b^2 + ab\delta_i \sin\theta_i - \delta_i \cos\theta_i c^2 - x_0}{\delta_i \cos^2\theta_i - 2\,\delta_i \cos^2\theta_i b^2 + 2\cos\theta_i ab\delta_i \sin\theta_i - \delta_i \cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0} \\
-\dfrac{\delta_i \sin\theta_i + b^2\delta_i \sin\theta_i + ba\delta_i \cos\theta_i - y_0}{\delta_i \cos^2\theta_i - 2\,\delta_i \cos^2\theta_i b^2 + 2\cos\theta_i ab\delta_i \sin\theta_i - \delta_i \cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0} \\
-\dfrac{-z_0 + ca\delta_i \cos\theta_i + cb\delta_i \sin\theta_i}{\delta_i \cos^2\theta_i - 2\,\delta_i \cos^2\theta_i b^2 + 2\cos\theta_i ab\delta_i \sin\theta_i - \delta_i \cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0} \\
\dfrac{\delta_i\left(-z_0 \sin\theta_i + ca\delta_i \cos\theta_i \sin\theta_i + cb\delta_i - cb\delta_i \cos^2\theta_i\right)}{\delta_i \cos^2\theta_i - 2\,\delta_i \cos^2\theta_i b^2 + 2\cos\theta_i ab\delta_i \sin\theta_i - \delta_i \cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0} \\
-\dfrac{\delta_i(-z_0 + ca\delta_i \cos\theta_i + cb\delta_i \sin\theta_i)\cos\theta_i}{\delta_i \cos^2\theta_i - 2\,\delta_i \cos^2\theta_i b^2 + 2\cos\theta_i ab\delta_i \sin\theta_i - \delta_i \cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0} \\
\dfrac{\delta_i\left(-\cos\theta_i \delta_i \sin\theta_i + 2\,\delta_i \cos\theta_i b^2 \sin\theta_i - ab\delta_i + 2\,ab\delta_i \cos^2\theta_i + \sin\theta_i \delta_i \cos\theta_i c^2 + \sin\theta_i x_0 - y_0 \cos\theta_i\right)}{\delta_i \cos^2\theta_i - 2\,\delta_i (\cos\theta_i)^2 b^2 + 2\cos\theta_i ab\delta_i \sin\theta_i - \delta_i \cos^2\theta_i c^2 - \cos\theta_i x_0 + \delta_i + b^2\delta_i - \sin\theta_i y_0}
\end{pmatrix}
$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0				
Revolute	0	1				
Prismatic/ Revolute	1	1				
Plane/plane contact	2	1				
Ball-and-socket	0	3				
Line/plane contact	2	2				
Line/sphere contact	1	3	a cylinder of radius R with axis $(a,b,c)^T$ passing by (x_0,y_0,z_0)	x_0, y_0, z_0 a, b, c R	δ_i	$\begin{pmatrix} a & c\,y_0 - b\,z_0 & 0 & 0 \\ b & a\,z_0 - c\,x_0 & 0 & 0 \\ c & b\,x_0 - a\,y_0 & 0 & 0 \\ 0 & a & 1 & 0 \\ 0 & b & 0 & 1 \\ 0 & c & 0 & 0 \end{pmatrix}$
Point/plane contact	2	3				

Type of the sensor: Matrix sensor

Type of the geometric feature: Edge

Elementary screw:

$$H_{i,j} = \begin{pmatrix} -\frac{a_1}{a_1\ \cos\theta_i\cos\phi_j+b_1\ \sin\theta_i\cos\phi_j+c_1\ \sin\phi_j} & -\frac{a_2}{a_2\ \cos\theta_i\cos\phi_j+b_2\ \sin\theta_i\cos\phi_j+c_2\ \sin\phi_j} \\ -\frac{b_1}{a_1\ \cos\theta_i\cos\phi_j+b_1\ \sin\theta_i\cos\phi_j+c_1\ \sin\phi_j} & -\frac{b_2}{a_2\ \cos\theta_i\cos\phi_j+b_2\ \sin\theta_i\cos\phi_j+c_2\ \sin\phi_j} \\ -\frac{c_1}{a_1\ \cos\theta_i\cos\phi_j+b_1\ \sin\theta_i\cos\phi_j+c_1\ \sin\phi_j} & -\frac{c_2}{a_2\ \cos\theta_i\cos\phi_j+b_2\ \sin\theta_i\cos\phi_j+c_2\ \sin\phi_j} \\ \frac{\delta_{i,j}(-\sin\theta_i\cos\phi_j c_1+\sin\phi_j b_1)}{a_1\ \cos\theta_i\cos\phi_j+b_1\ \sin\theta_i\cos\phi_j+c_1\ \sin\phi_j} & \frac{\delta_{i,j}(-\sin\theta_i\cos\phi_j c_2+\sin\phi_j b_2)}{a_2\ \cos\theta_i\cos\phi_j+b_2\ \sin\theta_i\cos\phi_j+c_2\ \sin\phi_j} \\ -\frac{\delta_{i,j}(\sin\phi_j a_1-\cos\theta_i\cos\phi_j c_1)}{a_1\ \cos\theta_i\cos\phi_j+b_1\ \sin\theta_i\cos\phi_j+c_1\ \sin\phi_j} & -\frac{\delta_{i,j}(\sin\phi_j a_2-\cos\theta_i\cos\phi_j c_2)}{a_2\ \cos\theta_i\cos\phi_j+b_2\ \sin\theta_i\cos\phi_j+c_2\ \sin\phi_j} \\ -\frac{\delta_{i,j}\cos\phi_j(\cos\theta_i b_1-\sin\theta_i a_1)}{a_1\ \cos\theta_i\cos\phi_j+b_1\ \sin\theta_i\cos\phi_j+c_1\ \sin\phi_j} & \frac{\delta_{i,j}\cos\phi_j(-\cos(\theta_i)b_2+\sin(\theta_i)a_2)}{a_2\ \cos\theta_i\cos\phi_j+b_2\ \sin\theta_i\cos\phi_j+c_2\ \sin\phi_j} \end{pmatrix}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0	the intersection of 2 planes	$\left\{\begin{array}{l} a_1, b_1, c_1, d_1 \\ a_2, b_2, c_2, d_2 \end{array}\right.$	$\delta_{i,j}$	$\begin{pmatrix} c_2\,b1 - b_2\,c_1 \\ a_2\,c_1 - c_2\,a_1 \\ b_2\,a_1 - b_1\,a_2 \\ 0 \\ 0 \\ 0 \end{pmatrix}$
Revolute	0	1				
Prismatic/ Revolute	1	1				
Plane/plane contact	2	1				
Ball-and-socket	0	3				
Line/plane contact	2	2				
Line/sphere contact	1	3				
Point/plane contact	2	3				

Type of the sensor: Matrix sensor

Type of the geometric feature: Plane

Elementary screw:

$$H_{i,j} = \begin{pmatrix} -\dfrac{a}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \\ -\dfrac{b}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \\ -\dfrac{c}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \\ \dfrac{\delta_{i,j}\left(-\sin\theta_i\cos\phi_j c + \sin\phi_j b\right)}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \\ -\dfrac{\delta_{i,j}\left(a\sin\phi_j - c\cos\theta_i\cos\phi_j\right)}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \\ \dfrac{\delta_{i,j}\cos\phi_j\left(a\sin\theta_i - b\cos\theta_i\right)}{a\cos\theta_i\cos\phi_j + b\sin\theta_i\cos\phi_j + c\sin\phi_j} \end{pmatrix}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0				
Revolute	0	1				
Prismatic/ Revolute	1	1				
Plane/plane contact	2	1	a plane	a, b, c, d	$\delta_{i,j}$	$\begin{pmatrix} 0 & a & 0 \\ c & 0 & 0 \\ -b & -c & 0 \\ 0 & 0 & a \\ 0 & 0 & b \\ 0 & 0 & c \end{pmatrix}$
Ball-and-socket	0	3				
Line/plane contact	2	2				
Line/sphere contact	1	3				
Point/plane contact	2	3				

Type of the sensor: Matrix sensor

Type of the geometric feature: Sphere

Elementary screw:

$$H_{i,j} = \begin{pmatrix} \dfrac{\delta_{i,j}\cos\theta_i\cos\phi_j - x_0}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \\ \dfrac{\delta_{i,j}\sin\theta_i\cos\phi_j - y_0}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \\ -\dfrac{-\delta_{i,j}\sin\phi_j + z_0}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \\ -\dfrac{\delta_{i,j}\left(\sin\theta_i\cos\phi_j z_0 - \sin\phi_j y_0\right)}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \\ \dfrac{\delta_{i,j}\left(-\sin\phi_j x_0 + \cos\theta_i\cos\phi_j z_0\right)}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \\ -\dfrac{\cos\phi_j\delta_{i,j}\left(\cos\theta_i y_0 - \sin\theta_i x_0\right)}{\cos\theta_i\cos\phi_j x_0 + \sin\theta_i\cos\phi_j y_0 - \delta_{i,j} + \sin\phi_j z_0} \end{pmatrix}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0				
Revolute	0	1				
Prismatic/ Revolute	1	1				
Plane/plane contact	2	1				
Ball-and-socket	0	3	a sphere	$(x_0, y_0, z_0), R$	$\delta_{i,j}$	$\begin{pmatrix} 0 & y_0 & z_0 \\ z_0 & -x_0 & 0 \\ -y_0 & 0 & x_0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$
Line/plane contact	2	2				
Line/sphere contact	1	3				
Point/plane contact	2	3				

Type of the sensor: Matrix sensor

Type of the geometric feature: Cylinder

Elementary screw:

$$H_{i,j} = \frac{1}{e} h_{i,j}$$

with

$$\begin{aligned} e \;=\; & cos^2\theta_i \cos^2 phi_j \delta_{i,j} - cos\theta_i \cos\phi_j x_0 \\ & +2\cos\theta_i \cos\phi_j ac\delta_{i,j} \sin\phi_j - 2\cos^2\theta_i \cos^2\phi_j \delta_{i,j} b^2 \\ & +2\cos\theta_i \cos^2\phi_j ab\delta_{i,j} \sin\theta_i - \cos^2\theta_i \cos^2\phi_j \delta_{i,j} c^2 \\ & -\sin\theta_i \cos\phi_j y_0 + \cos^2\phi_j b^2 \delta_{i,j} + 2\sin\theta_i \cos\phi_j bc\delta_{i,j} \sin\phi_j + \delta_{i,j} - \sin\phi_j z_0 \\ & +c^2\delta_{i,j} - c^2\delta_{i,j} \cos^2\phi_j \end{aligned}$$

et

$$\begin{aligned} h_{i,j}(1) \;=\; & -2\,\delta_{i,j} \cos\theta_i \cos\phi_j + x_0 - ac\delta_{i,j} \sin\phi_j \\ & +\delta_{i,j} \cos\theta_i \cos\phi_j b^2 - ab\delta_{i,j} \sin\theta_i \cos\phi_j + \delta_{i,j} \cos\theta_i \cos\phi_j c^2 \end{aligned}$$

$$h_{i,j}(2) = -\delta_{i,j} \sin\theta_i \cos\phi_j + y_0 - b^2\delta_{i,j} \sin\theta_i \cos\phi_j - bc\delta_{i,j} \sin\phi_j - ba\delta_{i,j} \cos\theta_i \cos\phi_j$$

$$h_{i,j}(3) = -\delta_{i,j} \sin\phi_j + z_0 - ca\delta_{i,j} \cos\theta_i \cos\phi_j - cb\delta_{i,j} \sin\theta_i \cos\phi_j - c^2\delta_{i,j} \sin\phi_j$$

$$\begin{aligned} h_{i,j}(4) \;=\; & \delta_{i,j}(\sin\theta_i \cos\phi_j z_0 - \sin\theta_i \cos^2\phi_j ca\delta_{i,j} \cos\theta_i - 2\,\cos^2\phi_j cb\delta_{i,j} + \cos^2\phi_j cb\delta_{i,j} \cos^2\theta_i \\ & -\sin\theta_i \cos\phi_j c^2\delta_{i,j} \sin\phi_j - \sin\phi_j y_0 + \sin\phi_j b^2\delta_{i,j} \sin\theta_i \cos\phi_j + bc\delta_{i,j} + \sin\phi_j ba\delta_{i,j} \cos\theta_i \cos\phi_j) \end{aligned}$$

$$\begin{aligned} h_{i,j}(5) \;=\; & -\delta_{i,j}(\sin\phi_j \delta_{i,j} \cos\theta_i \cos\phi_j - \sin\phi_j x_0 + ac\delta_{i,j} - ac\delta_{i,j} \cos^2\phi_j - \sin\phi_j \delta_{i,j} \cos\theta_i \cos\phi_j b^2 \\ & +\sin\phi_j ab\delta_{i,j} \sin\theta_i \cos\phi_j - 2\,\sin\phi_j \delta_{i,j} \cos\theta_i \cos\phi_j c^2 + \cos\theta_i \cos\phi_j z_0 \\ & -\cos^2\theta_i \cos^2\phi_j ca\delta_{i,j} - \cos\theta_i \cos^2 cb\delta_{i,j} \sin\theta_i) \end{aligned}$$

$$\begin{aligned} h_{i,j}(6) \;=\; & -\delta_{i,j} \cos\phi_j(-\cos\theta_i \delta_{i,j} \sin\theta_i \cos\phi_j - \cos\theta_i y_0 + 2\,\cos\theta_i b^2\delta_{i,j} \sin\theta_i \cos\phi_j + \cos\theta_i bc\delta_{i,j} \sin\phi_j \\ & +2\,ba\delta_{i,j} \cos^2\theta_i \cos\phi_j + \sin\theta_i x_0 - \sin\theta_i ac\delta_{i,j} \sin\phi_j - \delta_{i,j} \cos\phi_j ab + \sin\theta_i \delta_{i,j} \cos\theta_i \cos\phi_j c^2) \end{aligned}$$

Virtual Linkage	Class T	Class R	3 D pattern	3 D parameters	Sensor signals	Free space basis
Rigid	0	0				
Prismatic	1	0				
Revolute	0	1				
Prismatic/ Revolute	1	1	a cylinder	$x_0, y_0, z_0,$ a, b, c, R	$\delta_{i,j}$	$\begin{pmatrix} a & c\,y_0 - b\,z_0 \\ b & a\,z_0 - c\,x_0 \\ c & b\,x_0 - a\,y_0 \\ 0 & a \\ 0 & b \\ 0 & c \end{pmatrix}$
Plane/plane contact	2	1				
Ball-and-socket	0	3				
Line/plane contact	2	2				
Line/sphere contact	1	3				
Point/plane contact	2	3				

Bibliography

[AA93] Karim Ait-Abderrahim. *Commande de Robots Mobiles*. PhD thesis, Ecole Nationale Supérieure des Mines de Paris, January 1993.

[BF95] Jérôme Barraquand and Pierre Ferbach. Motion Planning with Uncertainty : The Information Space Approach. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 1341–1348, Nagoya, Japan, May 1995.

[Bui94] Xuân-Nam Bui. *Planification de Trajectoire pour un Robot Polygonal Non-Holonome dans un Environnement Polygonal*. PhD thesis, Ecole Nationale Supérieure des Mines de Paris, January 1994.

[Cha90] François Chaumette. *La Relation Vision-Commande: Théorie et Application à des Tâches Robotiques*. PhD thesis, Université de Rennes I, July 1990.

[CRE93] F. Chaumette, P. Rives, and B. Espiau. *Classification and Realization of Vision-Based Tasks*, pages 199–228. World Scientific Series in Robotics and Automated Systems, 1993.

[ECR90] Bernard Espiau, François Chaumette, and Patrick Rives. Une Nouvelle Approche de la Relation Vision-Commande en Robotique. Technical Report 1172, INRIA, March 1990.

[FH93] Armando Fox and Seth Hutchinson. Exploiting Visual Constraints in the Synthesis of Uncertainty-Tolerant Motion Plans. Part I: The Directional Backprojection and Part II: The Nondirectional Backprojection. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 305–316, 1993.

[Fra91] Th. Fraichard. Smooth Trajectory Planning for a Car in a Structured World. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 318–323, Sacramento, California, June 1991.

[FS94] Th. Fraichard and A. Scheuer. Car-Like Robots and Moving Obstacles. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 64–69, San Diego, California, 1994.

[Kha86] O. Khatib. Real-Time Obstacle Avoidance for Manipulators and Mobile Robots. *International Journal of Robotics Research*, 5(1):90–98, 1986.

[KL] P.J.S. Knightbridge and D. M. Lane. Preference Planning : A Reactive Approach to Robot Task Planning. To appear.

[KTB87] David J. Kriegman, Ernst Triendl, and Thomas O. Binford. A Mobile Robot: Sensing, Planning and Locomotion. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 402–408, 1987.

[Lat90] Jean-Claude Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1990.

[LS95] Vladimir J. Lumelsky and Andrei M. Shkel. Incorporating Body Dynamics into the Sensor-based Motion Planning Paradigm. The Maximum Turn Strategy. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 1637–1642, Nagoya, Japan, May 1995.

[Mas81] Matthew T. Mason. Compliance and Force Control for Computer Controlled Mainpulators. *IEEE Transactions on Systems, Man and Cybernetics*, June 1981.

[MI95] Jun Miura and Katsushi Ikeuchi. Generating Visual Sensing Strategies in Assembly Tasks. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 1912–1918, Nagoya, Japan, May 1995.

[Nov94] Cyril Novales. *Pilotage par Actions Réflexes et Navigation Locale de Robots Mobile Rapides*. PhD thesis, Université de Montpellier II, Octobre 1994.

[PG93] Roger Pissard-Gibollet. *Conception et Commande par Asservissement Visuel d'un Robot Mobile*. PhD thesis, Ecole Nationale Supérieure des Mines de Paris, Décembre 1993.

[PS95] Lance A. Page and Arthur C. Sanderson. Robot Motion Planning for Sensor-Based Control with Uncertainties. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 1333–1340, Nagoya, Japan, May 1995.

[SEB90] C. Samson, B. Espiau, and M. Le Borgne. *Robot Control: The Task Function Approach*. Oxford University Press, 1990.

[ZF92] Zenghyou Zhang and Olivier D. Faugeras. Estimation of Displacements from two 3D Frames Obtained from Stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1992.