GdR Robotics Winter School: Robotics Principia

Requirements for Sensor-based Control

François Chaumette

Inria

Univ Rennes, CNRS, IRISA, Rennes

Requirements

Basics in basic math and geometry:

- Linear algebra, pseudo-inverse, SVD, null space: see slides 3 to 5
- Changes of frames, rotations, velocity screw: see slides 6 to 8

Basics in robotics

- Geometric/kinematic robot model, robot Jacobian: see Modeling course

Basics in control

- P, PI, kinematic control: see Control course

Basic in sensors

- Sensor model and calibration: see Perception course
- Hand-“eye” calibration: see slides 9 to 14

(Moore-Penrose) Pseudo inverse

The $n \times m$ pseudo inverse $\mathbf{A}^+$ of any $m \times n$ matrix $\mathbf{A}$ is the only one matrix such that

$$\begin{cases} \mathbf{A}\mathbf{A}^+\mathbf{A} = \mathbf{A} \\ \mathbf{A}^+\mathbf{A}\mathbf{A}^+ = \mathbf{A}^+ \\ (\mathbf{A}\mathbf{A}^+)^T = \mathbf{A}\mathbf{A}^T \\ (\mathbf{A}^+\mathbf{A})^T = \mathbf{A}^+\mathbf{A} \end{cases}$$

Widely used for solving any (over/under)-constrained least-squares linear system:

If we look for vector $\mathbf{x}$ such that $||\mathbf{A}\mathbf{x} - \mathbf{b}||$ is minimal, then $\hat{\mathbf{x}} = \mathbf{A}^+\mathbf{b}$ is the only one solution such that $||\hat{\mathbf{x}}||$ is also minimal

Pseudo inverse

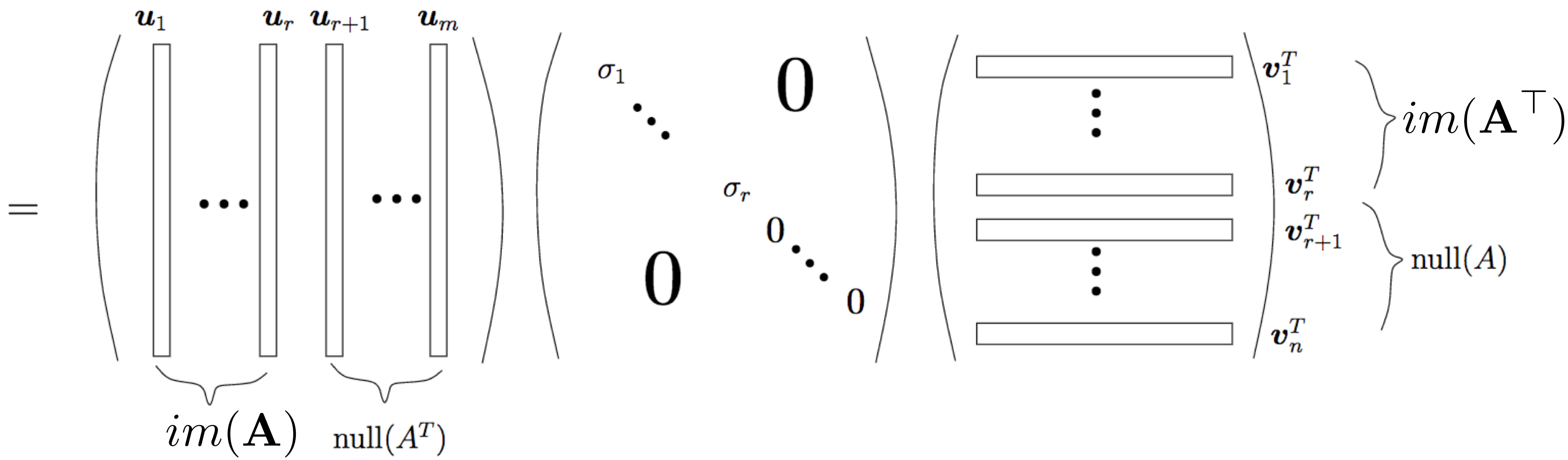
Let $r = \text{rank}\ (\mathbf{A})$

- If $m = n = r,\ \mathbf{A}^+ = \mathbf{A}^{-1}$
- If $m > n$ and $r = n,\ \mathbf{A}^+ = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T$
  In that case: $\mathbf{A}^+\mathbf{A} = \mathbf{I}_n$ (left inverse)
- If $m < n$ and $r = m,\ \mathbf{A}^+ = \mathbf{A}^T\left(\mathbf{A}\mathbf{A}^T\right)^{-1}$
  In that case: $\mathbf{A}\mathbf{A}^+ = \mathbf{I}_m$ (right inverse)
- In general, use the Singular Value Decomposition (SVD) $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$
  where $\mathbf{U}\mathbf{U}^T = \mathbf{I}_m,\ \mathbf{V}\mathbf{V}^T = \mathbf{I}_n$
  and $\mathbf{S}$ has 0 elements everywhere apart r values $\sigma_i \neq 0$ on its diagonal
  then $\mathbf{A}^+ = \mathbf{V}\mathbf{S}^+\mathbf{U}^T$
  where $\mathbf{S}^+$ is 0 everywhere apart r values $1/\sigma_i \neq 0$ on its diagonal

SVD decomposition

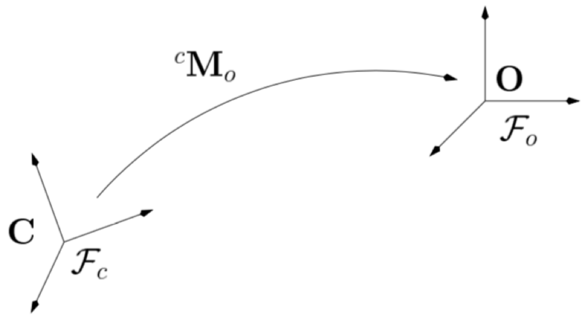
$$\underset{m \times n}{\mathbf{A}} = \underset{m \times m}{\mathbf{U}} \; \underset{m \times n}{\mathbf{S}} \; \underset{n \times n}{\mathbf{V}^\top}$$

$$\mathbf{V}^\top \mathbf{V} = \mathbf{I}_n \Rightarrow \begin{cases} \mathbf{v}_i^\top \mathbf{v}_i = 1 \; , \forall i \\ \mathbf{v}_i^\top \mathbf{v}_j = 0 \; , \forall i \neq j \end{cases}$$



Change of frames

pose $\mathbf{p} \in SE_3$



$$\mathbf{X}_c = {}^c\mathbf{R}_o\, \mathbf{X}_o + {}^c\mathbf{t}_o$$

$\mathbf{X}_c$: coordinates of $\mathbf{X}$ in $\mathcal{F}_c$
$\mathbf{X}_o$: coordinates of $\mathbf{X}$ in $\mathcal{F}_o$
${}^c\mathbf{t}_o$: position of $\mathbf{O}$ in $\mathcal{F}_c$
${}^c\mathbf{R}_o$: rotation matrix between $\mathcal{F}_c$ and $\mathcal{F}_o$

$$\mathbf{R} = \cos\theta\, \mathbf{I}_3 + \sin\theta\, [\mathbf{u}]_\times + (1-\cos\theta)\, \mathbf{u}\mathbf{u}^\top$$

$\mathbf{u}$: rotation axis ($\|\mathbf{u}\| = 1$) θ : rotation angle around $\mathbf{u}$

$[\mathbf{u}]_\times$: skew symetric matrix related to $\mathbf{u}$: $[\mathbf{u}]_\times = \begin{bmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{bmatrix}$

How to go from R to $\theta\mathbf{u}$

From Rodrigues formula

$$\mathbf{R} = \cos\theta\,\mathbf{I}_3 + \sin\theta\,\left[\mathbf{u}\right]_\times + (1-\cos\theta)\mathbf{u}\mathbf{u}^\top$$

we easily obtain

$$\begin{cases} \theta & = & \operatorname{Arccos}\left(\operatorname{tr}\mathbf{R} - 1\right)/2 \\ \\ \left[\theta\mathbf{u}\right]_\times & = & \frac{1}{2\ \operatorname{sinc}\ \theta}\left(\mathbf{R} - \mathbf{R}^\top\right) \end{cases}$$

where $\quad \operatorname{sinc}\ \theta = \dfrac{\sin\theta}{\theta} \qquad\qquad (\ \operatorname{sinc}\ 0 = 1)$

For $\quad \theta = \pi \quad \begin{cases} u_x & = & \sqrt{(1+r_{11})/2} \\ u_y & = & \sqrt{(1+r_{22})/2} \\ u_z & = & \sqrt{(1+r_{33})/2} \end{cases}$

Kinematic screw (instantaneous velocity)

$\mathbf{v} = (\boldsymbol{\upsilon}, \boldsymbol{\omega})$: kinematic screw between the camera and the scene

expressed at $\mathbf{C}$ in $\mathcal{F}_c$ (which is moving)

$\boldsymbol{\omega}$: rotational velocity : $\quad [\boldsymbol{\omega}]_\times = {}^o\mathbf{R}_c^\top \, {}^o\dot{\mathbf{R}}_c = -{}^o\dot{\mathbf{R}}_c^\top \, {}^o\mathbf{R}_c$

$\boldsymbol{\upsilon}$: translational velocity at $\mathbf{C}$: $\quad \boldsymbol{\upsilon}(\mathbf{O}) = -\boldsymbol{\upsilon}(\mathbf{C}) - \boldsymbol{\omega} \times \mathbf{CO}$

To express $\mathbf{v}$ at $\mathbf{O}$ in $\mathcal{F}_o$: ${}^o\mathbf{v} = {}^o\mathbf{V}_c \, \mathbf{v}$ with ${}^o\mathbf{V}_c = \begin{bmatrix} {}^o\mathbf{R}_c & [{}^o\mathbf{t}_c]_\times {}^o\mathbf{R}_c \\ \mathbf{0}_3 & {}^o\mathbf{R}_c \end{bmatrix}$

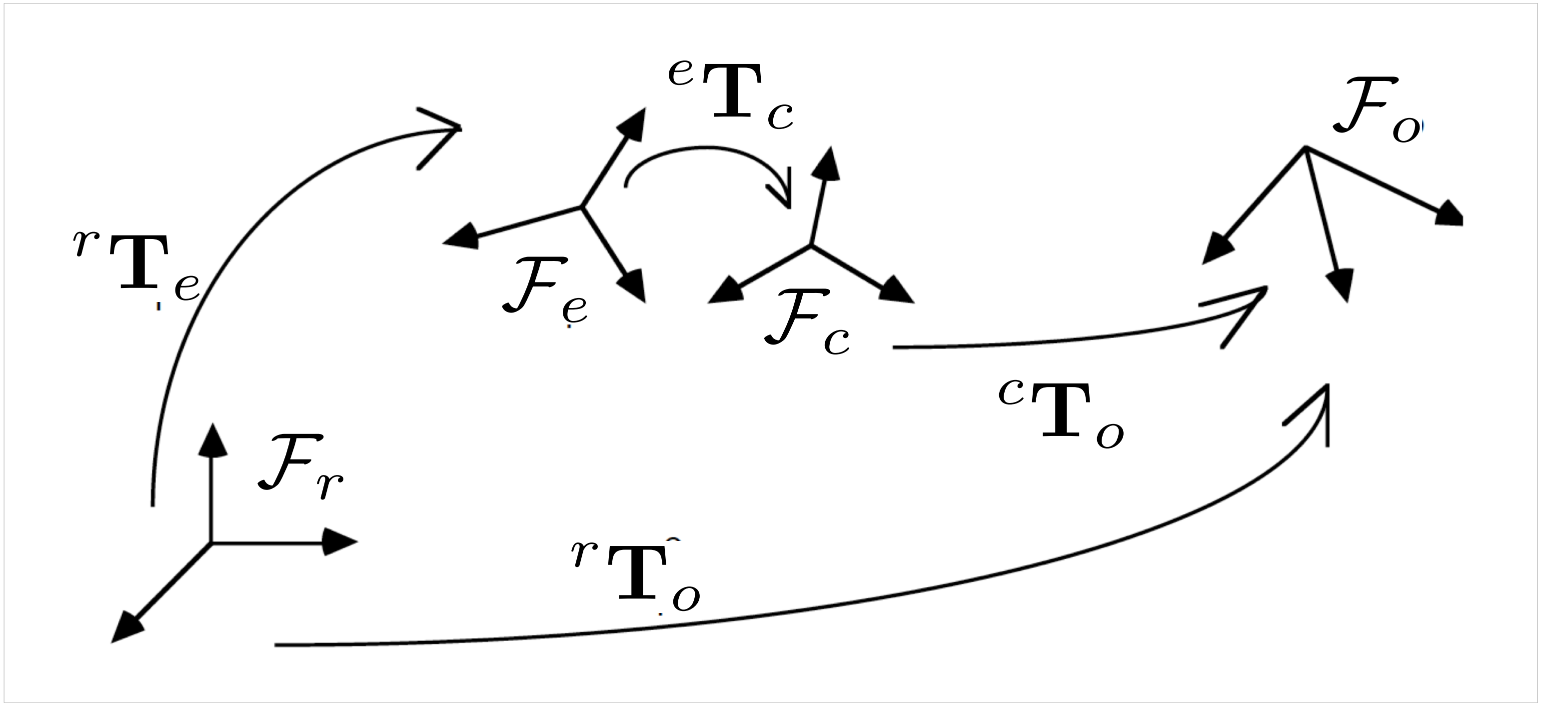
We can decompose $\mathbf{v}$ as $\mathbf{v} = \mathbf{v}_c - \mathbf{v}_o$

where $\mathbf{v}_c$: camera kinematic screw, expressed at $\mathbf{C}$ in $\mathcal{F}_c$

$\mathbf{v}_o$: object kinematic screw, expressed at $\mathbf{C}$ in $\mathcal{F}_c$

Hand-eye calibration

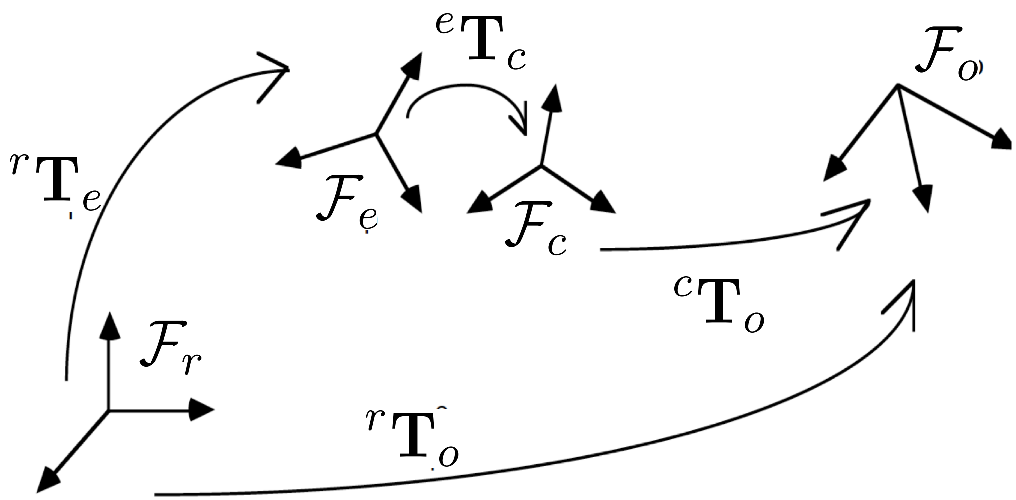
How to estimate the pose of a camera wrt. a robot?
(similar problem and method when the sensor is not a camera)



Eye-in-hand configuration

Hand-eye calibration

$$
\begin{aligned}
{}^{e}\mathbf{T}_c &= {}^{e}\mathbf{T}_r\ {}^{r}\mathbf{T}_o\ {}^{o}\mathbf{T}_c \\
&= {}^{r}\mathbf{T}_e^{-1}\ {}^{r}\mathbf{T}_o\ {}^{c}\mathbf{T}_o^{-1}
\end{aligned}
$$



- ${}^{e}\mathbf{T}_c$: the (constant) pose we are looking for
- ${}^{r}\mathbf{T}_e$: obtained from robot geometric model
- ${}^{c}\mathbf{T}_o$: obtained from pose estimation
- ${}^{r}\mathbf{T}_o$: constant but unknown

Idea: ${}^{r}\mathbf{T}_o$ is the same whatever the robot/camera pose, so with 2 poses i and j:

$$
\begin{aligned}
{}^{r}\mathbf{T}_o &= {}^{r}\mathbf{T}_{e_i}\ {}^{e_i}\mathbf{T}_{c_i}\ {}^{c_i}\mathbf{T}_o &= {}^{r}\mathbf{T}_{e_j}\ {}^{e_j}\mathbf{T}_{c_j}\ {}^{c_j}\mathbf{T}_o \\
&= {}^{r}\mathbf{T}_{e_i}\ {}^{e}\mathbf{T}_c\ {}^{c_i}\mathbf{T}_o &= {}^{r}\mathbf{T}_{e_j}\ {}^{e}\mathbf{T}_c\ {}^{c_j}\mathbf{T}_o
\end{aligned}
$$

since $\quad {}^{e}\mathbf{T}_c = {}^{e_i}\mathbf{T}_{c_i} = {}^{e_j}\mathbf{T}_{c_j}$

Hand-eye calibration

$$
\begin{array}{rrcl}
 & {}^{r}\mathbf{T}_{e_i}\;{}^{e}\mathbf{T}_{c}\;{}^{c_i}\mathbf{T}_{o} & = & {}^{r}\mathbf{T}_{e_j}\;{}^{e}\mathbf{T}_{c}\;{}^{c_j}\mathbf{T}_{o} \\
\Leftrightarrow & {}^{r}\mathbf{T}_{e_j}^{-1}\;{}^{r}\mathbf{T}_{e_i}\;{}^{e}\mathbf{T}_{c} & = & {}^{e}\mathbf{T}_{c}\;{}^{c_j}\mathbf{T}_{o}\;{}^{c_i}\mathbf{T}_{o}^{-1} \\
\Leftrightarrow & {}^{e_j}\mathbf{T}_{e_i}\;{}^{e}\mathbf{T}_{c} & = & {}^{e}\mathbf{T}_{c}\;{}^{c_j}\mathbf{T}_{c_i} \\
\Leftrightarrow & \mathbf{A}\;\mathbf{X} & = & \mathbf{X}\;\mathbf{B}
\end{array}
$$

Then, decompose the rotation and translation part:

$$
\left\{
\begin{array}{lcl}
{}^{e_j}\mathbf{R}_{e_i}\;{}^{e}\mathbf{R}_{c} & = & {}^{e}\mathbf{R}_{c}\;{}^{c_j}\mathbf{R}_{c_i} \\
\left({}^{e_j}\mathbf{R}_{e_i} - \mathbf{I}_3\right)\;{}^{e}\mathbf{t}_{c} & = & {}^{e}\mathbf{R}_{c}\;{}^{c_j}\mathbf{t}_{c_i} - {}^{e_j}\mathbf{t}_{e_i}
\end{array}
\right.
$$

Once ${}^{e}\mathbf{R}_{c}$ is known, ${}^{e}\mathbf{t}_{c}$ is obtained by solving a simple linear system

$\left({}^{e_j}\mathbf{R}_{e_i} - \mathbf{I}_3\right)$ is of rank 2 ; at least a third orientation k is necessary to obtain a full rank system with couples (i,j), (i,k) and (j,k)

Hand-eye calibration

$$\begin{array}{ccc} {}^{e}\mathbf{R}_{c} & : & \theta\mathbf{u} \\ {}^{e_j}\mathbf{R}_{e_i} & : & \theta_e\mathbf{u}_e \\ {}^{c_j}\mathbf{R}_{c_i} & : & \theta_c\mathbf{u}_c \end{array}$$

Thanks to rotation properties

${}^{e_j}\mathbf{R}_{e_i}\ {}^{e}\mathbf{R}_{c} = {}^{e}\mathbf{R}_{c}\ {}^{c_j}\mathbf{R}_{c_i}$ equivalent to linear system:

$$[\theta_c\mathbf{u}_c + \theta_e\mathbf{u}_e]_\times \tan\frac{\theta}{2}\mathbf{u} = \theta_c\mathbf{u}_c - \theta_e\mathbf{u}_e$$

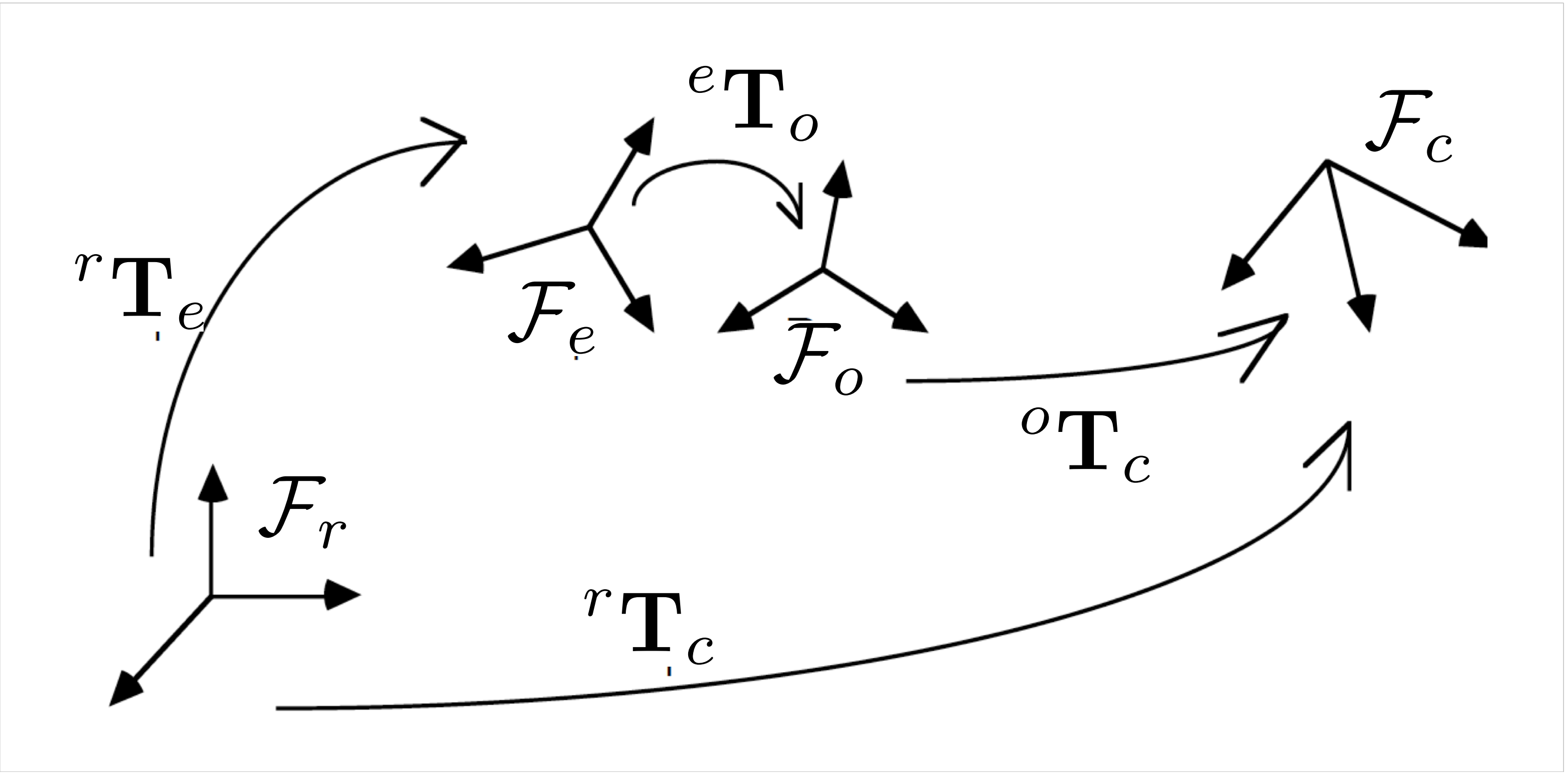
$[\mathbf{v}]_\times$: anti-symetric matrix of **v** such that $[\mathbf{v}]_\times\,\mathbf{u} = \mathbf{v}\times\mathbf{u}$

of rank 2 ; at least a third orientation k is necessary

Once $\theta\mathbf{u}$ is known, ${}^{e}\mathbf{R}_{c}$ is known (Rodrigues formula)

$$^{e}\mathbf{R}_{c} = \cos\theta\ \mathbf{I}_3 + \sin\theta\ [\mathbf{u}]_\times + (1-\cos\theta)\ \mathbf{u}\mathbf{u}^\top$$

Hand-eye calibration: eye-to-hand configuration



Similar problem: $ {}^{r}\mathbf{T}_{c} = {}^{r}\mathbf{T}_{e_i}\, {}^{e}\mathbf{T}_{o}\, {}^{c}\mathbf{T}_{o_i}^{-1} = {}^{r}\mathbf{T}_{e_j}\, {}^{e}\mathbf{T}_{o}\, {}^{c}\mathbf{T}_{o_j}^{-1}$

Similar resolution

References

Hand-eye calibration:

- *R. Tsai, R. Lenz*: **A new technique for fully autonomous efficient 3-D robotics hand-eye calibration**, IEEE Transactions on Robotics & Automation, 5(3): 345–358, June1989.

Implementation (C++ source code):

- ViSP: *http://visp.inria.fr*