ITROB1 - Robotics 1

Peter Ahrendt

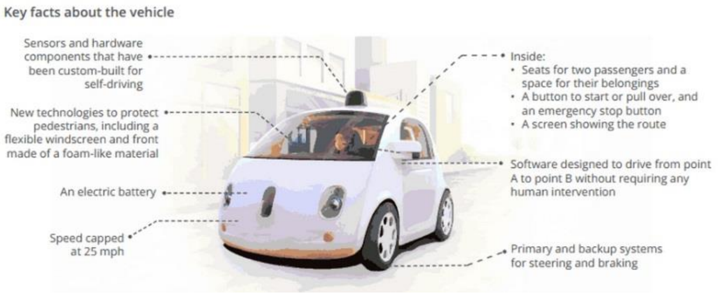
Course Intro

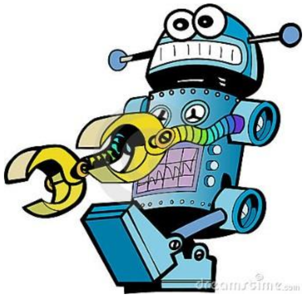
- Lecturers
  - Peter Ahrendt and Mads Dyrmann
- Mondays 12:00 – 16:00
  - Robotics theory + Matlab exercises
  - Robotic software development + Crustcrawler exercises
- Group exam based on final report (12-grade scale)
  - 30 min. oral defence (robot demo + presentation + questions)
- Three mandatory assignments to attend exam

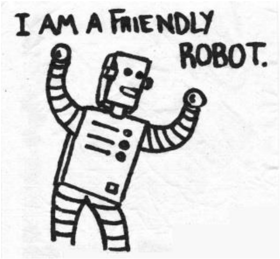
Course content

- Robotics theory
  - Coordinate transforms, kinematics, control, ..
- ROS (Robot Operating System)
- Hands-on robot programming
  - Python and Matlab
- Computer vision and robotics sensors
- Project-based exam
  - Crustcrawler + ROS + vision + ..

What is robotics ?







Robots

- One definition :

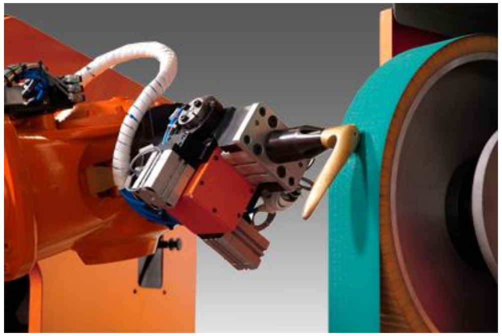
  "A goal-oriented machine that can

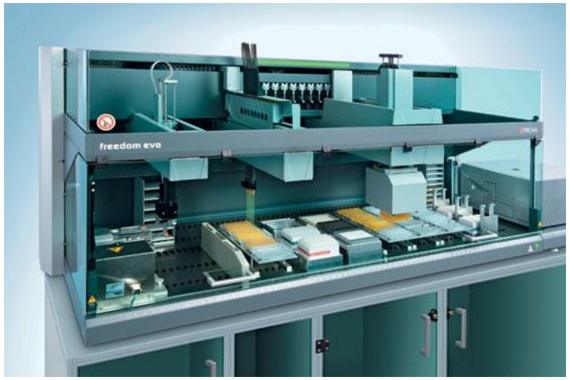
  <u>sense, plan and act</u>"

- Robota – (slave) work, hard work
  - Dirty, Dangerous and Dull

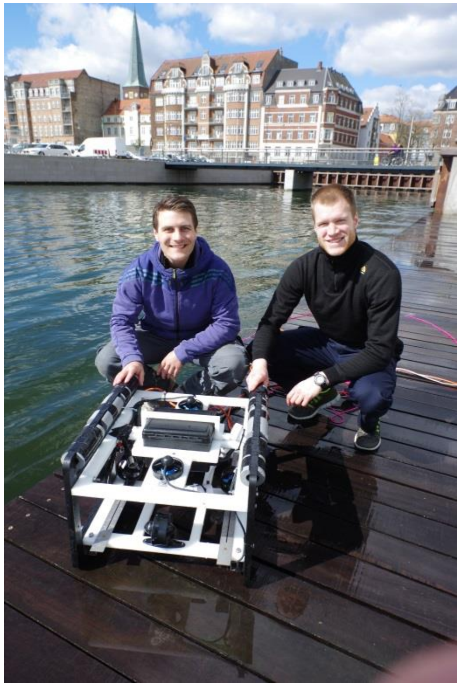
Industrial robots

Application examples

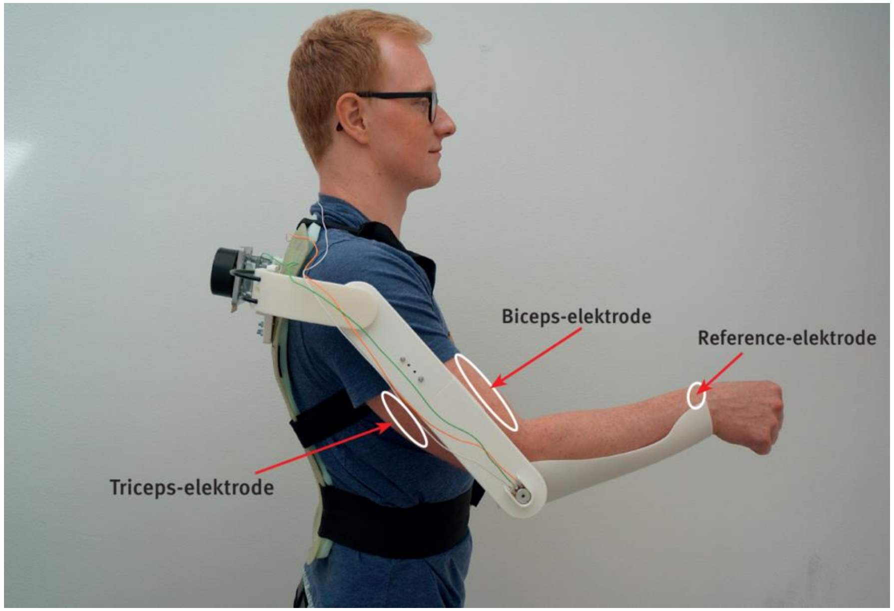


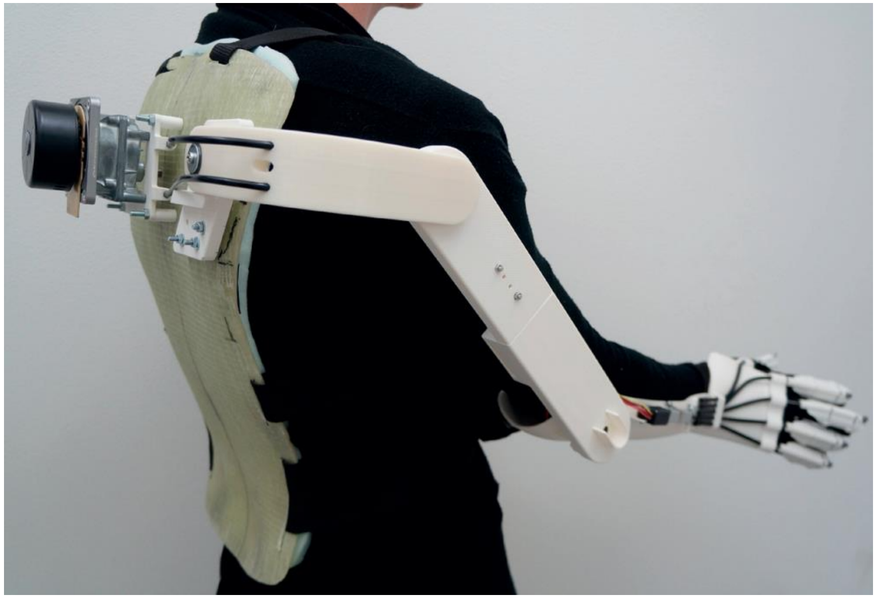


Autonomous Underwater Vehicle (AUV)



Exoskeleton





CrustCrawler

- http://www.crustcrawler.com/

Example final projects

- Locate + pick-and-place
- Parts assembly
- Motion planning using manual guiding
- Combinations of sensor/actuator technology + robot programming/simulation/theory
  - Kinect or Force sensor (eg. Flexiforce)
- Note:
  - Vision has to be in-the-loop
  - Has to use ROS node abstraction (at least 2 nodes)

Robotic systems overview

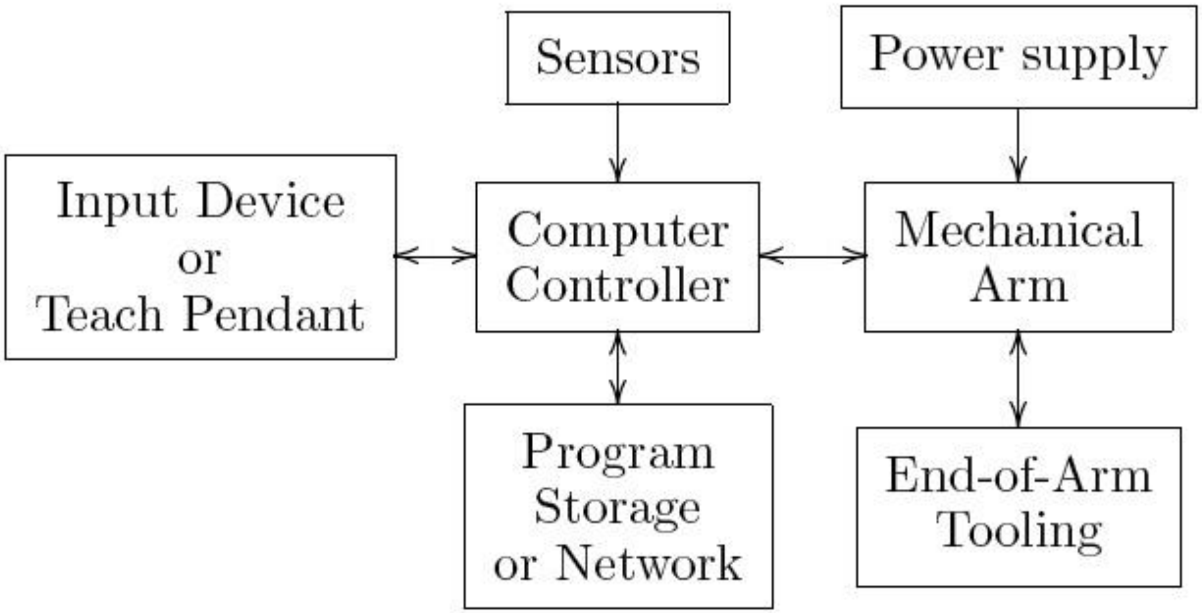
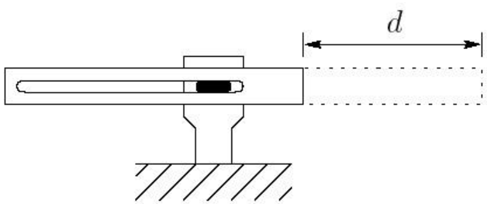
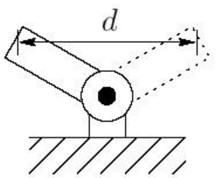


Figure 1.4: The integration of a mechanical arm, sensing, computation, user interface and tooling forms a complex robotic system. Many modern robotic systems have integrated computer vision, force/torque sensing, and advanced programming and user interface features.

Kinematic structures



Prismatic (P)



Revolute (R)

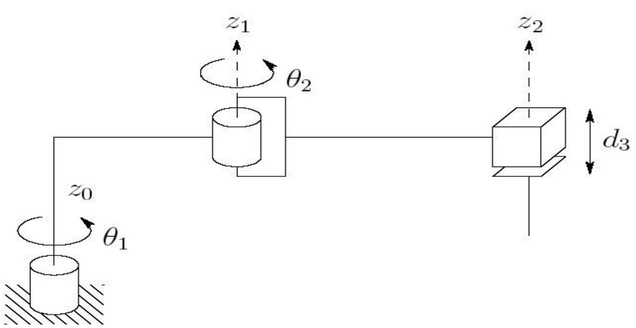
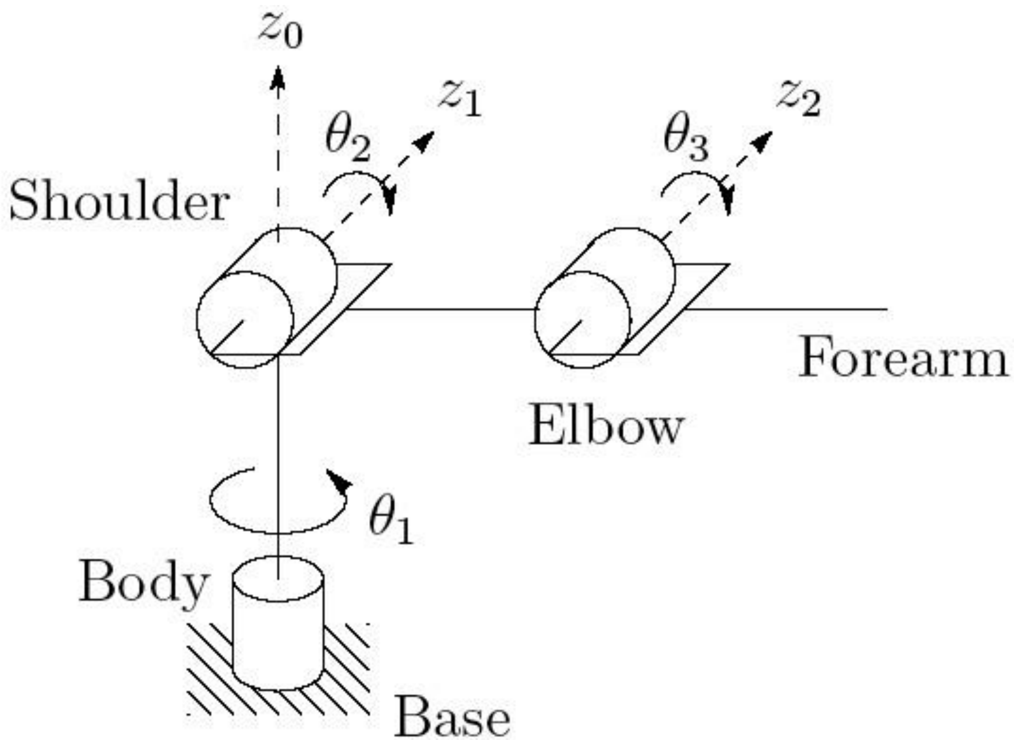
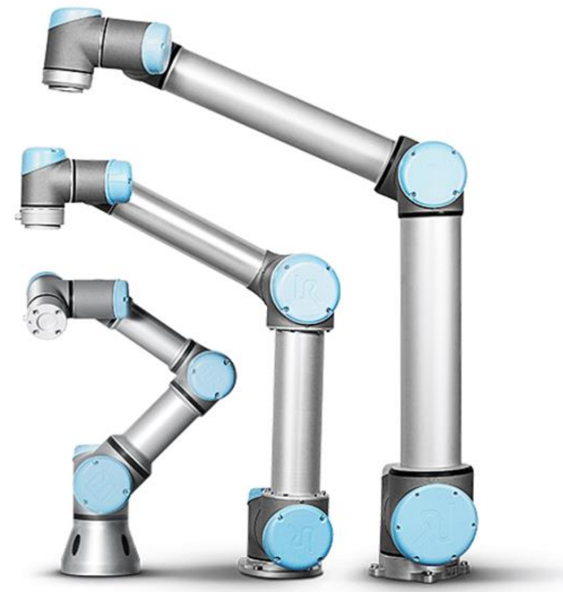


Figure 1.13: Symbolic representation of the SCARA arm.

SCARA – (RRP)

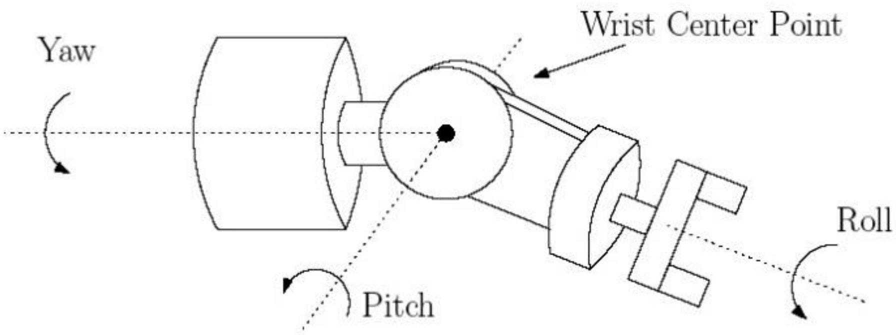
Kinematic structure





Articulated Manipulator (RRR)

Spherical wrist



Kinematic structures

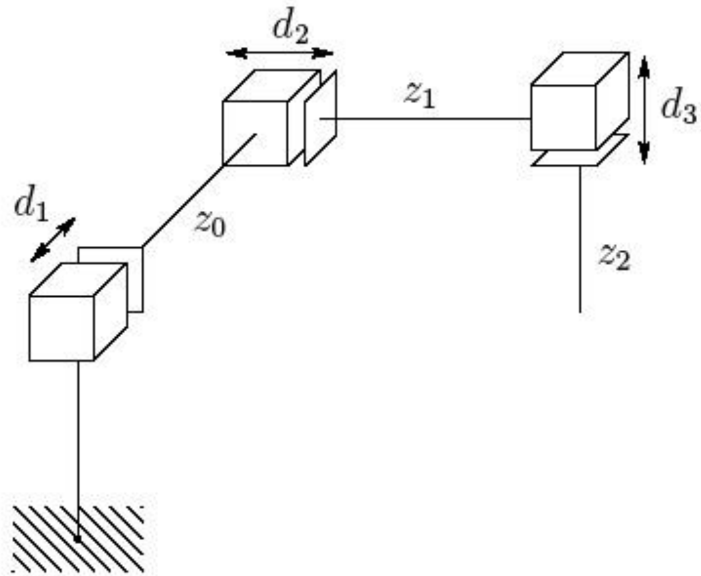
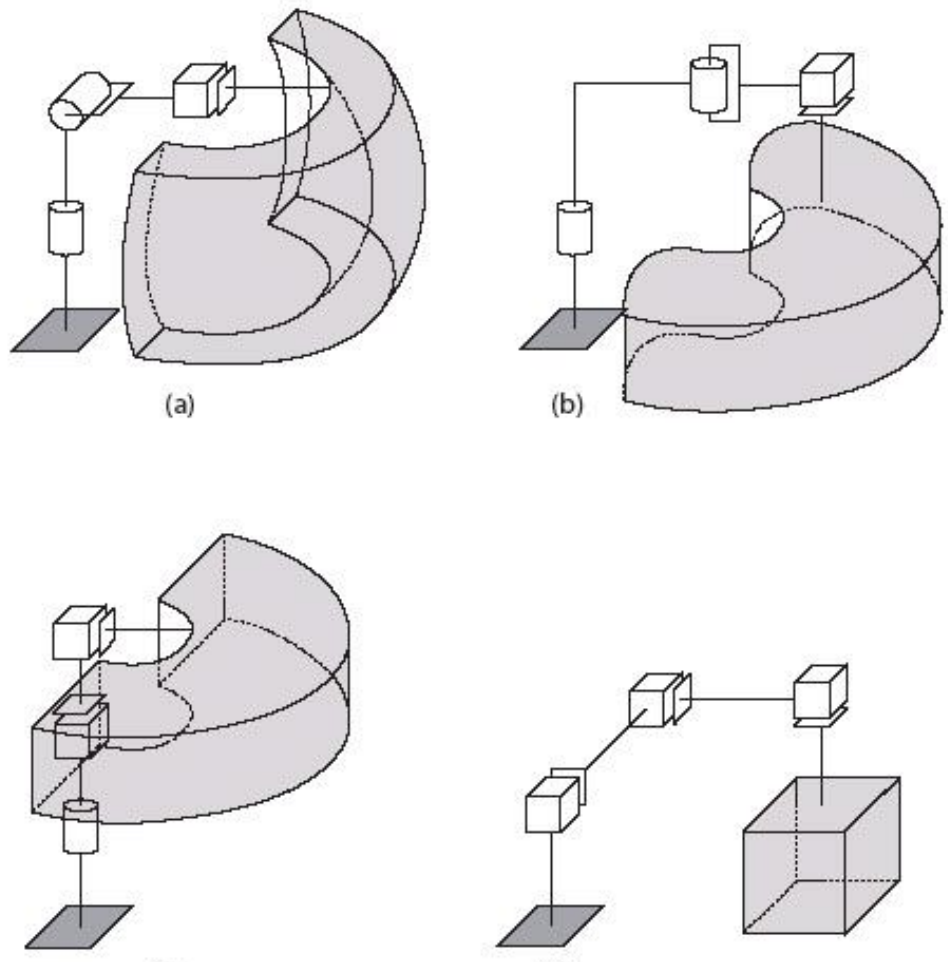


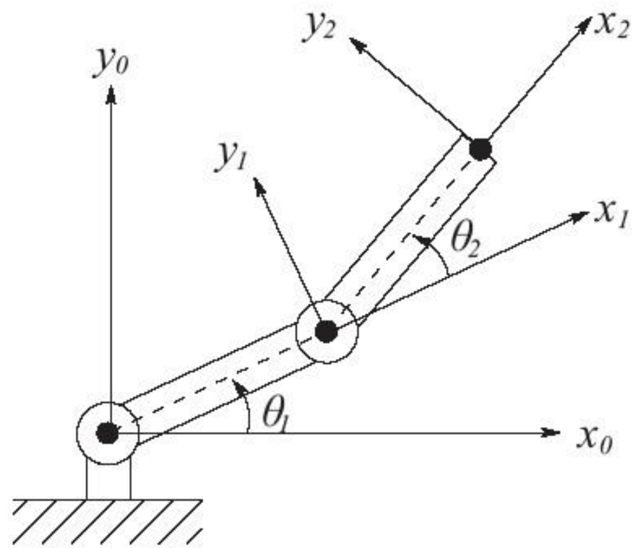
Figure 1.16: The Epson Cartesian Robot. Cartesian robot designs allow increased structural rigidity and hence higher precision. Cartesian robots are often used in pick and place operations. (Photo courtesy of Epson Robots.)

Workspace



(a) (b) (c) (d)

Degrees-of-freedom (DOF)



Forward and inverse kinematics

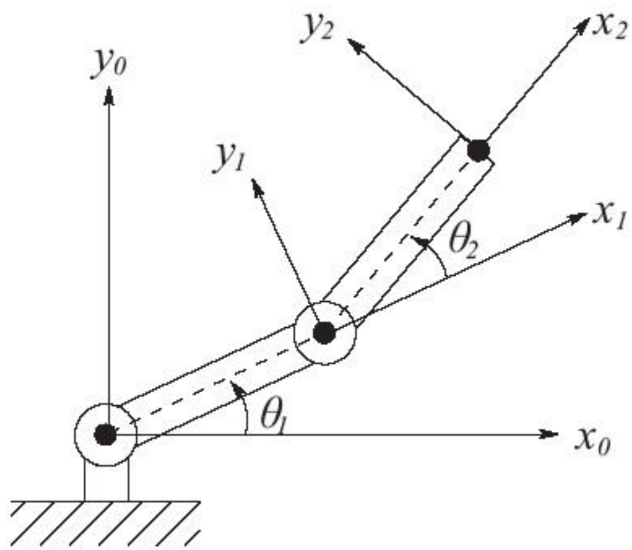
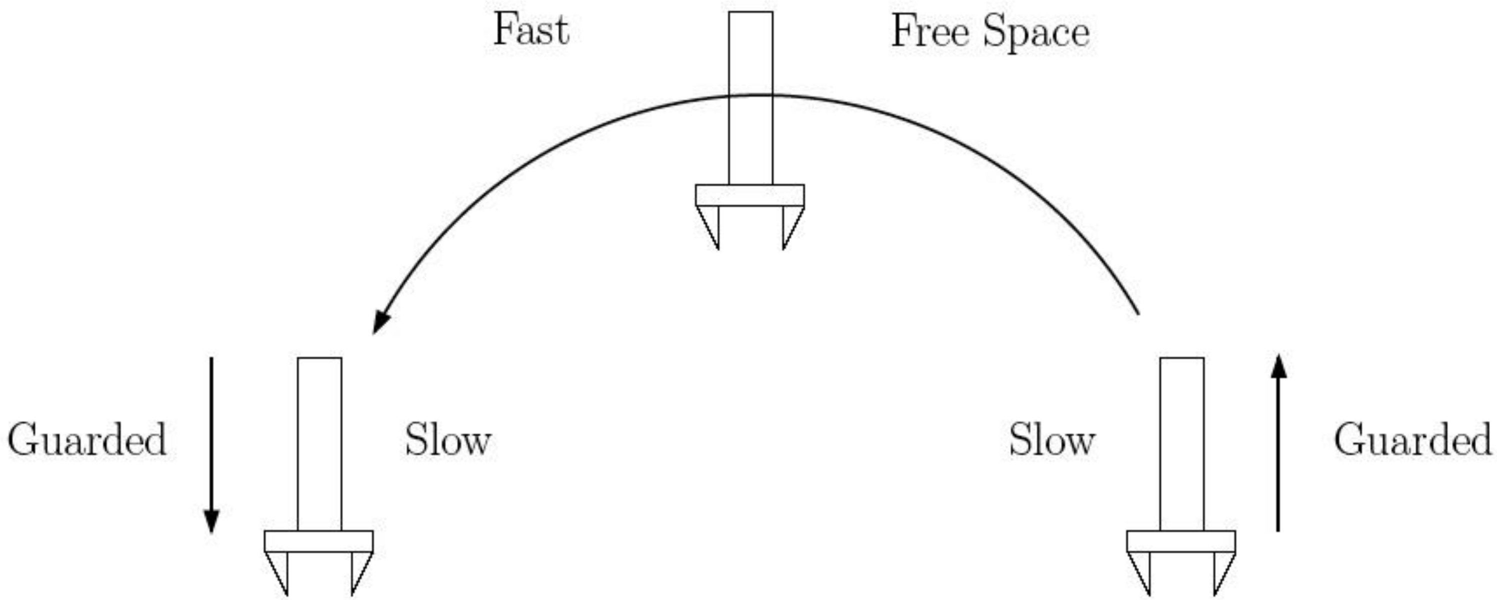


Figure 1.20: Coordinate frames attached to the links of a two-link planar robot. Each coordinate frame moves as the corresponding link moves. The mathematical description of the robot motion is thus reduced to a mathematical description of moving coordinate frames.

Robot planning



Force sensors / control

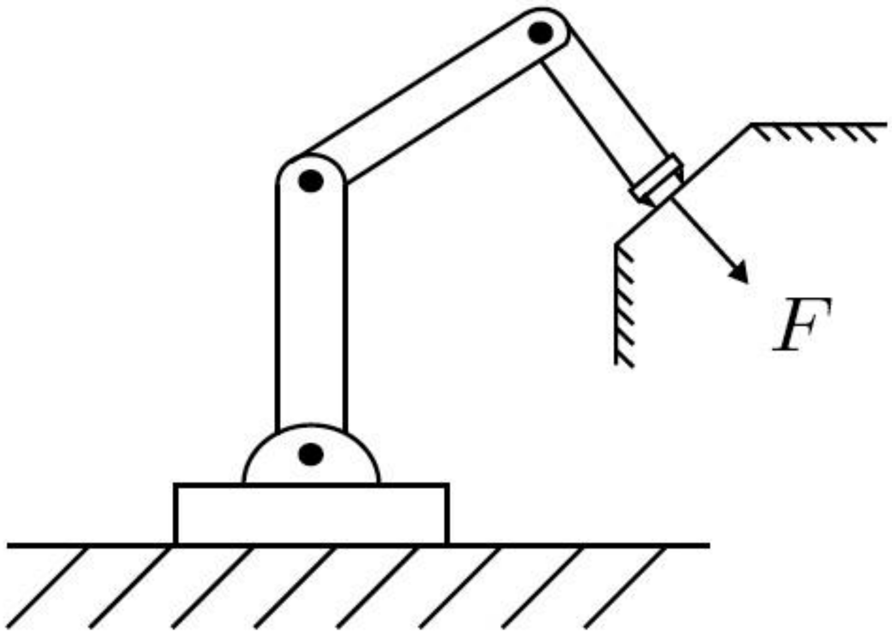
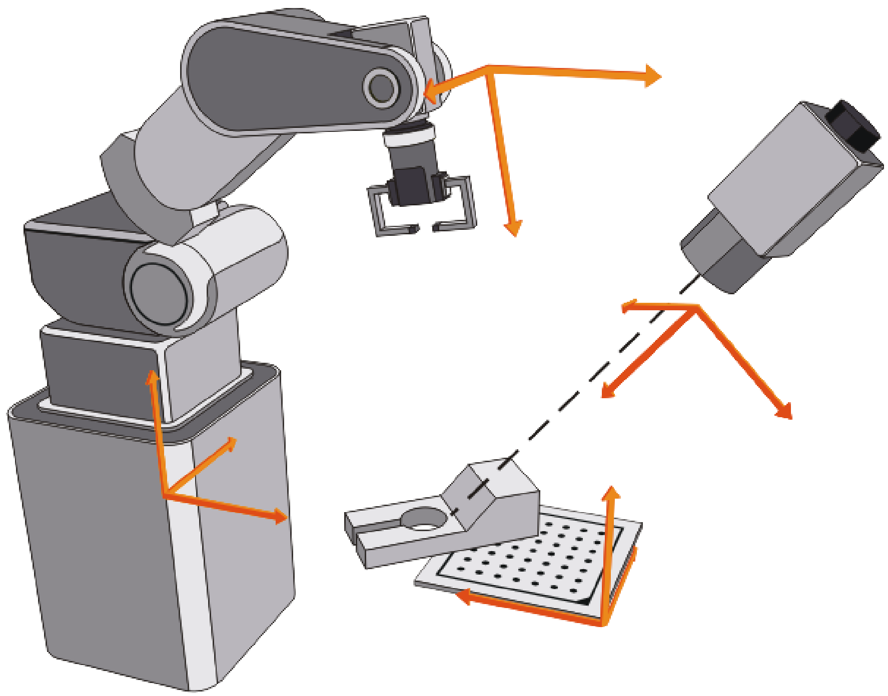


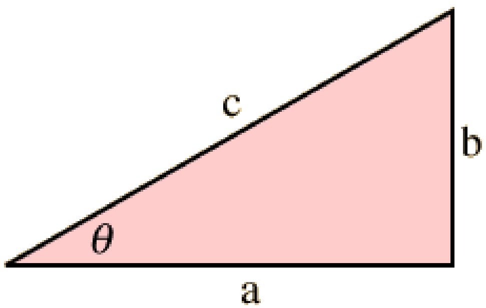
Figure 9.2: Robot end effector in contact with a rigid surface.

Robotic vision systems



Rigid motions and homogeneous transformations

Angles – cosine/sine



$$\text{sine}: \sin\theta = \frac{b}{c} = \frac{\textit{side opposite } \theta}{\textit{hypotenuse}}$$

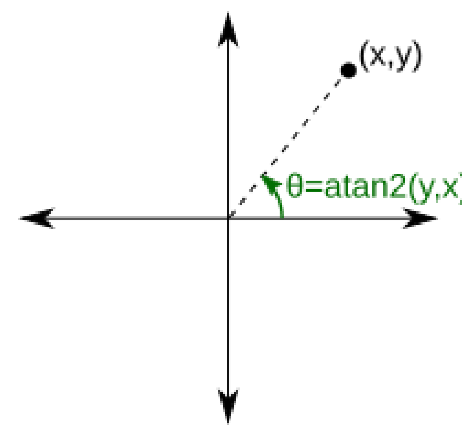
$$\text{cosine}: \cos\theta = \frac{a}{c} = \frac{\textit{side adjacent } \theta}{\textit{hypotenuse}}$$

$$\text{tangent}: \tan\theta = \frac{b}{a} = \frac{\textit{side opposite } \theta}{\textit{side adjacent } \theta} = \frac{\sin\theta}{\cos\theta}$$

$$\text{cotangent}: \cot\theta = \frac{\cos\theta}{\sin\theta} = \frac{1}{\tan\theta}$$

$$\text{secant}: \sec\theta = \frac{1}{\cos\theta}$$

$$\text{cosecant}: \csc\theta = \frac{1}{\sin\theta}$$



Atan2 function

Vectors

- E.g. p = [1, 2] (or sometimes $\vec{p}$)
- Can represent a point / position
  - .. or a translation
  - (and many other things)
- Can be added (parallellogram-rule)
  - eg. p1=[1,2], p2=[3,5] and p1+p2 = [4, 7]

Vectors for translation

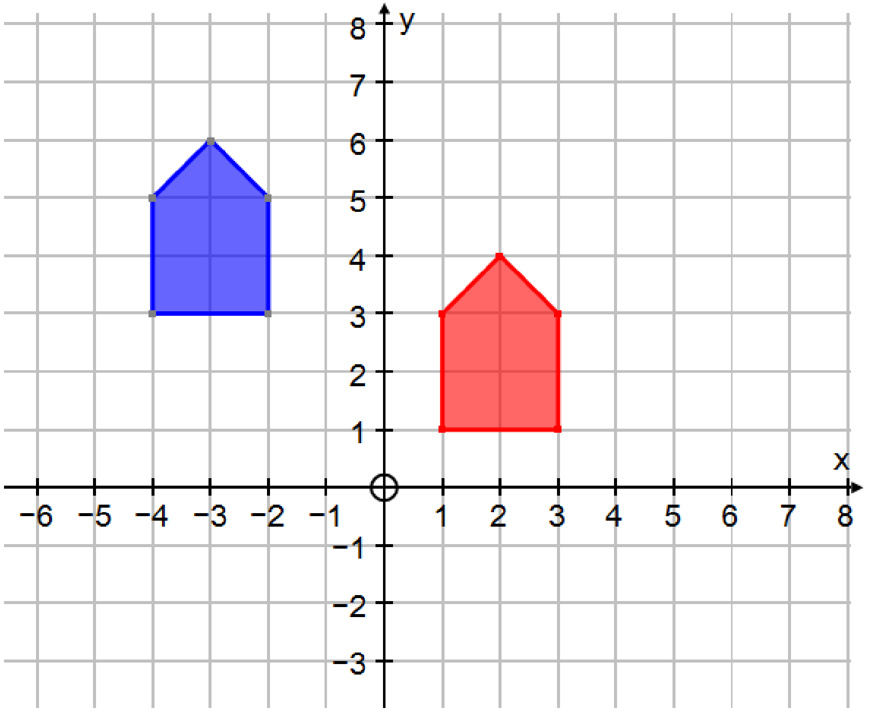
The blue object has been translated by what vector to produce the red image?

A. $\begin{pmatrix}3\\0\end{pmatrix}$

B. $\begin{pmatrix}5\\-2\end{pmatrix}$

C. $\begin{pmatrix}5\\2\end{pmatrix}$

D. $\begin{pmatrix}-2\\5\end{pmatrix}$

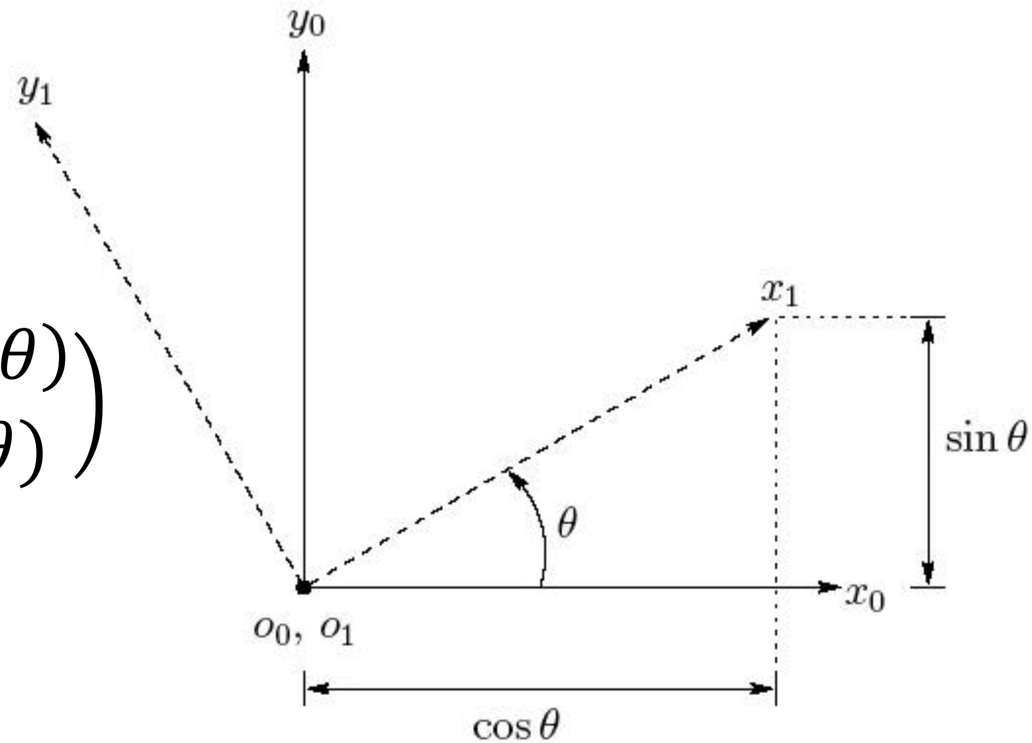


Matrices

- Eg. $\mathrm{R} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$
- Can represent rotation
  - (and many other things..)
- Can be multiplied with vector
  - E.g p = $\begin{pmatrix} 5 \\ 10 \end{pmatrix}$ then Rp = $\begin{pmatrix} 25 \\ 55 \end{pmatrix}$
- Or other matrix..
  - $\mathrm{R}_1 = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \mathrm{R}_2 = \begin{pmatrix} 2 & 4 \\ 7 & 9 \end{pmatrix}$ and $\mathrm{R}_1\mathrm{R}_2$ = $\begin{pmatrix} 16 & 22 \\ 34 & 48 \end{pmatrix}$
- Multiplication order is important!

Matrices for rotation

- $R = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ sin(\theta) & \cos(\theta) \end{pmatrix}$



- E.g. $\theta = \frac{\pi}{2}$ = 90 degrees, $R = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$
  - When $p_1 = \begin{pmatrix} 5 \\ 10 \end{pmatrix}$ then $p_0 = R\, p_1 = \begin{pmatrix} -10 \\ 5 \end{pmatrix}$

Rigid motions

- Rigid motion = translation + rotation
- Eg. $p_0 = R^0{}_1\, p_1 + d_0$
  - d_0 = translation and $R^0{}_1$ = rotation between o_0 and o_1
  - p_1 is coordinates of point in frame o_1 and p_0 is the coordinates of same point in frame o_0

Example : (2D) Planar robot arm

- Joints, Links and Frames + Rigid motion

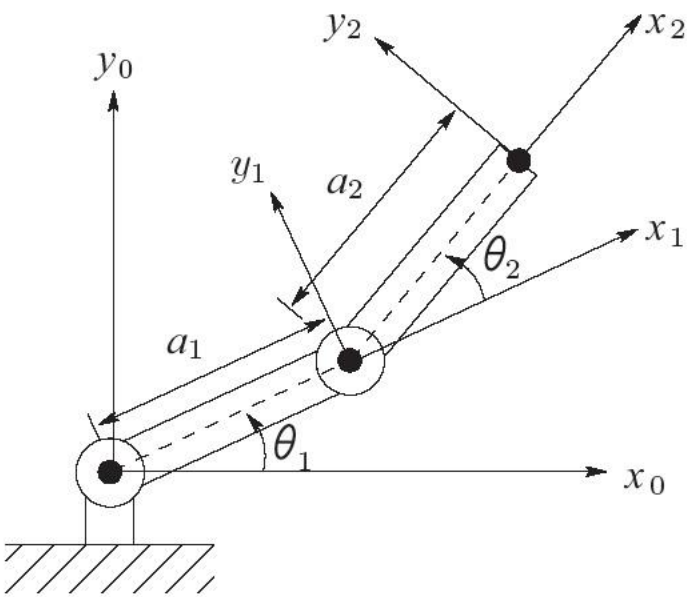


Figure 3.6: Two-link planar manipulator. The z-axes all point out of the page, and are not shown in the figure.

Homogeneous Transformation

- Combination of translation and rotation
- In 2D : H = $\begin{pmatrix} R & d \\ 0 & 1 \end{pmatrix}$
- .. or explicit H = $\begin{pmatrix} \cos(\theta) & -\sin(\theta) & d_x \\ \sin(\theta) & \cos(\theta) & d_y \\ 0 & 0 & 1 \end{pmatrix}$
- For example H = $\begin{pmatrix} 0 & -1 & 2 \\ 1 & 0 & 3 \\ 0 & 0 & 1 \end{pmatrix}$ = ”translate to [2,3] and rotate 90 degrees”

Homogeneous Representation

- Generalization of a point
- $P = \begin{pmatrix} p \\ 1 \end{pmatrix}$ where p is a point
  - Eg. $P_0 = \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix}$ or $P_1 = \begin{pmatrix} 15 \\ 18 \\ 1 \end{pmatrix}$

Using homogeneous transformations

- Calculating rigid motions
  - $P_0 = H\ P_1 = \begin{pmatrix} R^0{}_1 & d_0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} p_1 \\ 1 \end{pmatrix} = R^0{}_1\ p_1 + d_0$
- For example, finding end effector position
  - ie. $P_1 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$ and then $P_0 = H\ P_1 = d_0$

Kinematic chains and transformation matrices

- Complete transformation from frame 0 to frame n is given by

$$T^0_n = A^0_1 \, A^1_2 \, .. \, A^{n-1}_n$$

(The A-matrices are the homogeneous transformation matrices between frames)

- Similar results between other frames..
  - Eg. finding the origin of joint 2 in a 4-joint robot would just use $T^0_2 = A^0_1 \, A^1_2$

Exercises

- Form groups of 3-4 persons (ideally multi-disciplinary!)

- Week 1 Exercises and Assignment
  - (MANDATORY) Robot example assignment
  - Exercise : 2D Planar Robot in MATLAB

- Install :
  - MATLAB
  - Robotics Toolbox (from petercorke.com)