Introduction: Outline & Logistics

EEL 4930/5934: Autonomous Robots

Spring 2023

Md Jahidul Islam

Lecture 1

ECE | Florida
Electrical & Computer Engineering
UF | UNIVERSITY of FLORIDA

EEL 4930/5934: Autonomous Robots

⇒ **Spring 2023 Lectures**

- M,W,F | LAR 0330 | 3:00-3:50 PM
- Final exam: May 3rd at 12:30-2:30 PM

⇒ **Course Directory:** Canvas

⇒ **Forum:** Teams

AuRo: Autonomous Robots

Robotics and intelligent systems

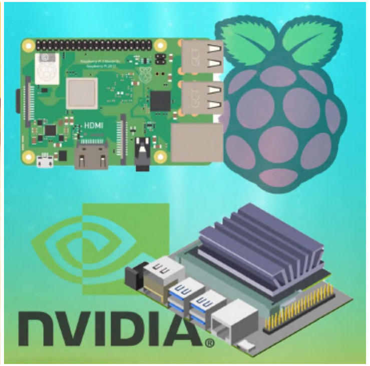
⇒ Domain: ground, underwater, aerial

⇒ Topic: perception, planning, control

⇒ Skill: hardware, software, domain

⇒ Theory and experimental fundamentals

- Sensory design and integration concepts
- Robot Operating System (ROS)
- Motion kinematics analytics
- Robot perception, planning, and control
- Unmodeled environmental/social factors
- Aspects of field deployments



⇒ **Hw platforms:** TurtleBot-4, J. Nanos, R. Pis

⇒ **OS platforms:** ROS / ROS2 (Python / C++)

AuRo Resources

⇒ ***Textbooks***

- ***Introduction to Robotics: Mechanics and Control*** *By John Craig.*

  ISBN-13: 978-0133489798. Pearson; 4th edition.

- ***Probabilistic Robotics*** *By Sebastian Thrun, Wolfram Burgard and Dieter Fox.*

  ISBN-13: 978-0262201629, Intelligent Robotics & Autonomous Agents; 1st Edition.

Check if they are available online!

⇒ Recommended pointers

- Robot Operating System (ROS): http://wiki.ros.org/ROS/Tutorials
- ROS2 tutorials: https://docs.ros.org/en/foxy/Tutorials.html
- **Good online resources:**
  - Robotics Specialization by UPenn; AI for Robotics from GTech
  - MIT 18.06 Linear Algebra; SLAM Course by Prof. Cyrill Stachniss
  - TUM Multiple View Geometry by Prof. Daniel Cremers
  - UC Berkeley Artificial Intelligence CS 188 by Prof. Pieter Abbeel

Class lecture materials and handouts are very important

Instructor, TA, and Rubric

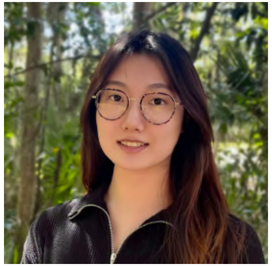
Instructor

Md Jahidul Islam

- **Email**: jahid@ece.ufl.edu
- **OH**: Thursday 4:00 - 5:00 PM

  At LAR 339D

TA



Boxiao Yu

- **Email:** boxiao.yu@ufl.edu
- **OH:** Tuesday 4:00-5:00 PM

  At NEB 222

Item	Points	% of Final Grade
Hands-on Homework (HH1 - HH4)	4 x 10	40
Analytical Homework (AH1 - AH2)	2 x 10	20
Mid-term	1 x 15	15
Final exam / project option	1 x 25	25
Total		***100***

Percent	Grade	Grade Points
92 or More	A	4.00
88.0 - 91.9	A-	3.67
84.0 - 87.9	B+	3.33
81.0 - 83.9	B	3.00
78.0 - 80.9	B-	2.67
75.0 - 77.9	C+	2.33
72.0 - 74.9	C	2.00
67.0 - 71.9	C-	1.67
64.0 - 66.9	D+	1.33
60.0 - 63.9	D	1.00
55.0 - 59.9	D-	0.67
Below 55	E	0.00

EEL 4930 vs EEL 5934

⇒ Pre-requisites

- One of
  - EEL 4744C: Microprocessor Applications
  - EEL4712C: Digital Design
- Fluent in object-oriented programming
  - Python and/or C++
- Basics of linear algebra and calculus
- +EEL 5934: Embedded systems or AI/ML

⇒ HH: Hands-on Homework

- Four HHs: 4 x 10 = 40
  - **HH1:** ROS integration and simulation
  - **HH2:** Kinematics
  - **HH3:** Visual Perception
  - **HH4:** Active Planner
- +EEL 5934: some advanced parts

⇒ AH: Analytical Homework

- Two AHs: 2 x 10 = 20
  - **AH1:** Kinematics
  - **AH2:** SLAM
- +EEL 5934: some advanced parts

⇒ Mid-term and Final/Project

- Mid-term exam: 15 points
  - Approx 9th week (will be announced)
  - Paper-based (most likely open-book)
- Final exam or project: 25 points
  - Finals are paper-based individual exam
  - Project option is strongly encouraged
    - Details are due **by March 22**
    - Team members: 1-5
- More details will be discussed in class

AuRo: Topics Overview (1/3)

⇒ **Week 1-2: Course Introduction and Logistics**

- Robotics and AI overview: past, present, and future
- ROS overview: ROS/ROS2 for robotics

⇒ **Week 2-3: Robot Systems Components and Sensory Integration**

- Electronics and computational platforms
- Sensory integration and design choices
- User interface, Integration, Middleware (ROS)

HH1 is Out

⇒ **Week 3-4: Robot Systems Components and Sensory Integration**

- Locomotion: ground robots (UGVs)
- 2-DOF and 3-DOF robots
- Forward and inverse kinematic

Some advanced topics for EEL 5934

AH1 is Out

⇒ **Week 4-5: Locomotion: Underwater and Aerial Robots**

- 5-DOF and 6-DOF robots
- Dynamics of thrusters and propellers

Some advanced topics for EEL 5934

HH1 Due; HH2 Out

AuRo: Topics Overview (2/3)

⇒ **<u>Week 6-7:</u> Perception: UGVs / AUVs / UAVs**

- 2D and 2D perception
- Visual and inertial measurements
- Gyroscope, accelerometers, IMU, GPS
- Range sensors, camera vision and LiDAR
- Acoustic and hyperspectral perception

AH1 Due

⇒ **<u>Week 8-9:</u> Deep Diving Into Visual Perception**

- Robot vision basics - UGVs / AUVs / UAVs
- Image processing and filtering
- Object detection, tracking, and following
- Stereo vision and 3D perception

Some advanced topics for EEL 5934

HH2 Due; HH3 Out

Mid-term By Wk-10

⇒ **<u>Week 10-11:</u> Planning: Motion Planners and Path Planners**

- Dynamic programming and SOTA planners
- UGV and UAV planning: social and behavioral aspects
- AUV planning: semi-autonomous and long-term planning

Project Finalization

HH3 Due; HH4 Out

AuRo: Topics Overview (3/3)

⇒ **Week 12-13: SLAM: Simultaneous Localization and Mapping**

- 2-DOF, 3-DOF, and 6-DOF robots
- VInS (Visual inertial SLAM), SVinS
- Acoustic and optical localization

Some advanced topics for EEL 5934

HH4 Due; AH2 Out

⇒ **Week 13-14: Control: Linear and Non-linear Controls**

- Kalman Filtering (KF), extended KF, unscented KF
- Particle filters and probabilistic filtering
- Summary discussions: UGVs / AUVs / UAVs

Some advanced topics for EEL 5934

AH2 Due

⇒ **Week 15: Final Exam and Project Showcase**

Project Updates

AuRo: General Expectations

⇒ Understanding the past, present, and future of autonomous robotics

⇒ Getting exposed to the current state-of-the-art of scientific literature

⇒ Developing hands-on experience in the relevant engineering aspects (in HHs and AHs)

⇒ Trying out some ideas or extensions of your own through a final project

⇒ This course should prepare you well for becoming a roboticist

- Competitive robotics laboratories in academia / industry

⇒ Time and effort commitment: high (so is the reward)

⇒ Tools you should be comfortable with:

- Basic hw/sw systems and linear algebra concepts
- Python/C++ programing!

Make hundred mistakes - just don't repeat the same mistakes

AuRo: Grading Policy

⇒ Grading and reporting

- Grades are periodically posted online
- Grades are final after 1 week of posting
- Checkout the general UF grading policy

Item	Points	% of Final Grade
Hands-on Homework (HH1 - HH4)	4 x 10	40
Analytical Homework (AH1 - AH2)	2 x 10	20
Mid-term	1 x 15	15
Final exam / project option	1 x 25	25
Total		***100***

⇒ Final rubric and grade assignment

- HH/AH grades are final once graded
- Mid-term, and final/project grades will be curved if necessary
- **Top 3** projects will get **+5** bonus points!

Percent	Grade	Grade Points
92 or More	A	4.00
88.0 - 91.9	A-	3.67
84.0 - 87.9	B+	3.33
81.0 - 83.9	B	3.00
78.0 - 80.9	B-	2.67
75.0 - 77.9	C+	2.33
72.0 - 74.9	C	2.00
67.0 - 71.9	C-	1.67
64.0 - 66.9	D+	1.33
60.0 - 63.9	D	1.00
55.0 - 59.9	D-	0.67
Below 55	E	0.00

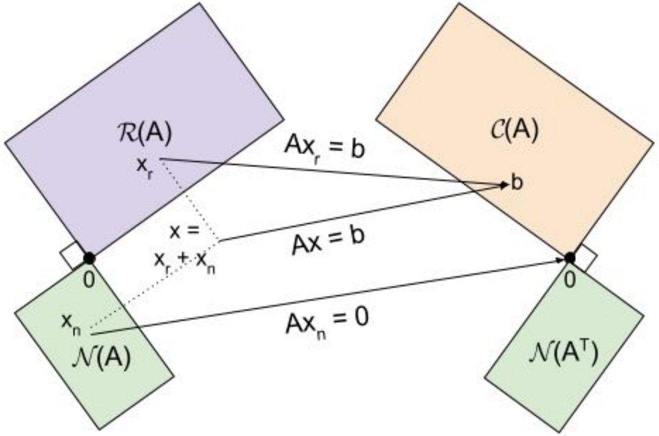
⇒ *Pay attention to the feedback and be present!*

⇒ *Attending lectures is crucial in this course*

Self Assessment (1/2)

$$A = \begin{bmatrix} 3 & 4 & -2 \\ 1 & 4 & -1 \\ 2 & 6 & -1 \end{bmatrix}$$

- Does $\mathbf{A}^{-1}$ exist?
- What are the eigenvalues of A?
- What is the rank of A?



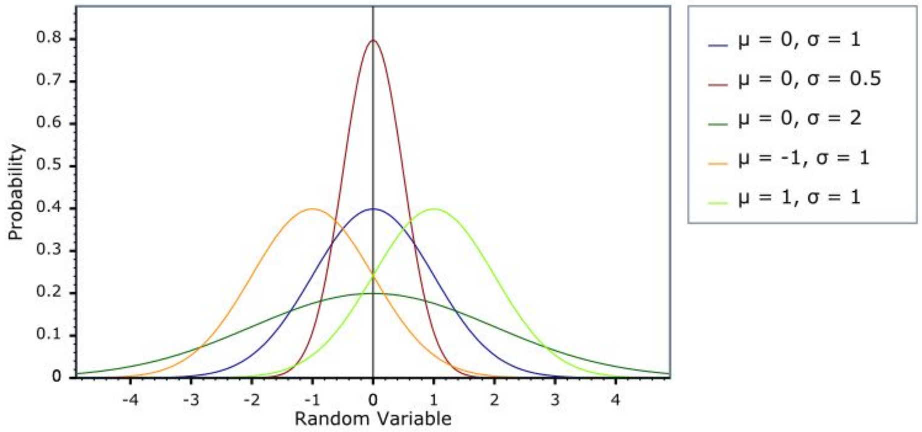
- What is a vector space?
- What are the four fundamental subspaces?
- What is, and how to do
  - LU factorization
  - SVD: Singular Value decomposition

How to solve **Ax=b** when

- **A** is square and invertible/non-invertible
- **A** is rectangular (**mxn**): m>n or m<n

- Matrix and vector norms

Self Assessment (2/2)



$$p(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

What is ***p(x; μ, Σ)*** for multivariate Gaussian (normal)?

$$p(x; \mu, \Sigma) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$$

- What is a probability distribution?
- What are:
  - PMF: probability mass function
  - PDF: probability density function
  - CDF: cumulative distribution function

Write a Python/C++ code for probabilistic sampling

- Coin toss
- Gaussian distribution

Lecture Contents

⇒ Robot, robotics, and AI

⇒ Major **types** and **kinds** of robots/robotics

- Cartesian coordinate robots
- Wheeled robots
- Robot manipulators
- Surgical robots
- Assistive and Social Robots
- Legged and humanoid robots
- Underwater robots
- Aerial robots
- Micro-robots and soft robotics
- …

⇒ Robot Components

⇒ A brief history of robotics

⇒ Getting started in ROS

- ROS-1 (melodic or noetic)
- ROS-2
- Rviz

What Is A Robot?

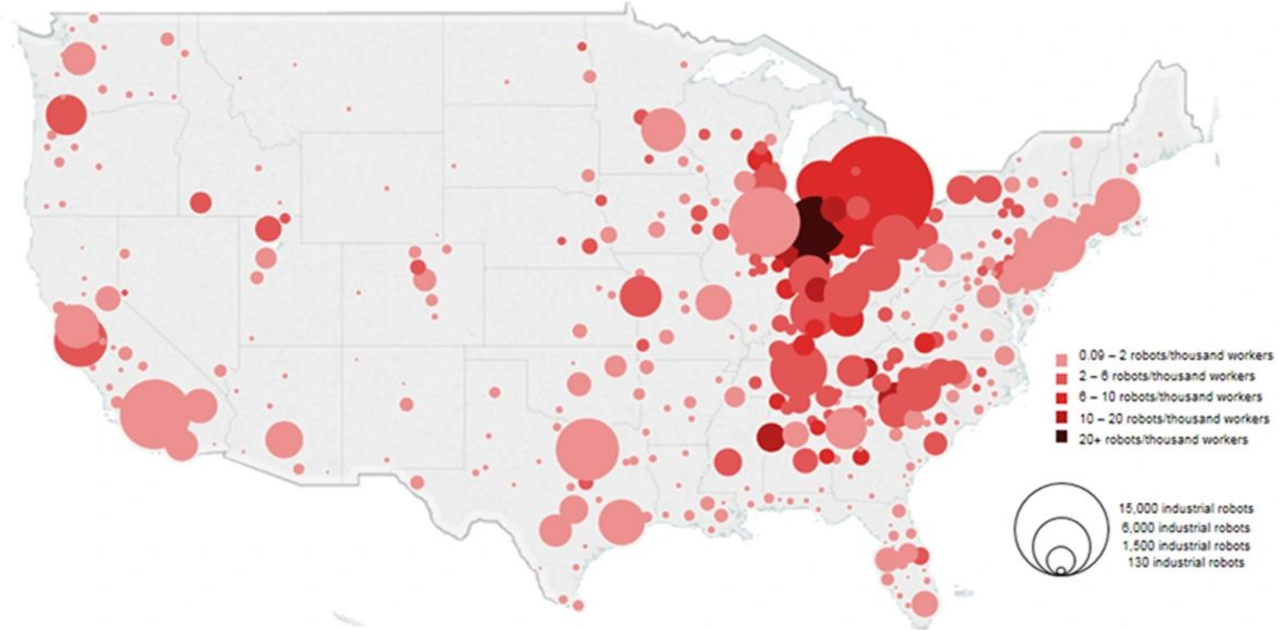
⇒ A machine (system or subsystem)

- Designed for specific tasks
- Programmable
- Does on-board computation

⇒ Autonomous or semi-autonomous
⇒ Mobile or stationary
⇒ Standalone or collaborative
⇒ *Artificially Intelligent or non-intelligent!*

Where Are The Robots?

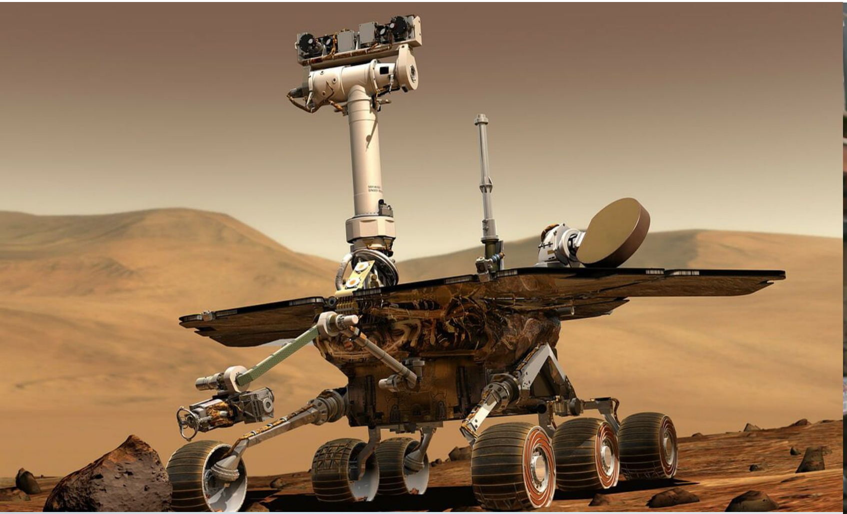
Number and incidence of industrial robots (per thousand workers) by metropolitan statistical area, 2015



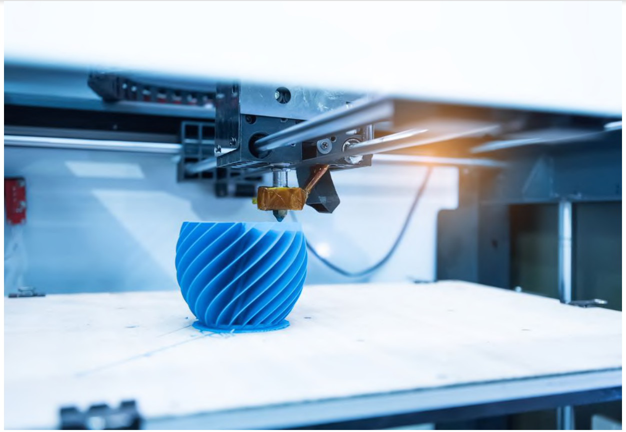
Note: Robot incidence reflects private employment only
Source: Brookings analysis of International Federation of Robotics data

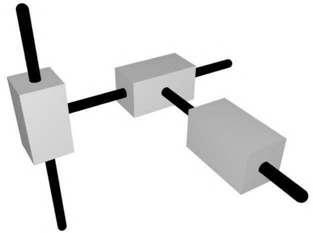
B | Metropolitan Policy Program
at BROOKINGS

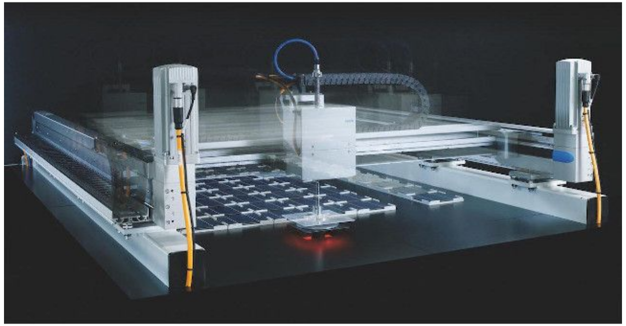
Autonomous Robots And Systems



Cartesian Coordinate Robots



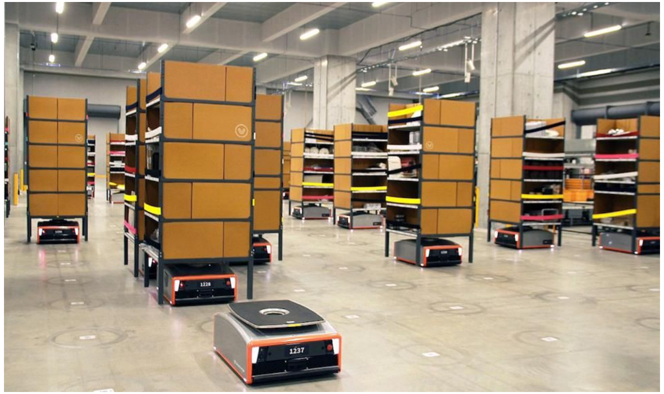




⇒ Cartesian Robots

- Also called *linear robots*
- Three principal axes with linear control
- Can cover a 3D (2.5D) Space
- <u>Example:</u> 3D printers, scanners, and painting robots

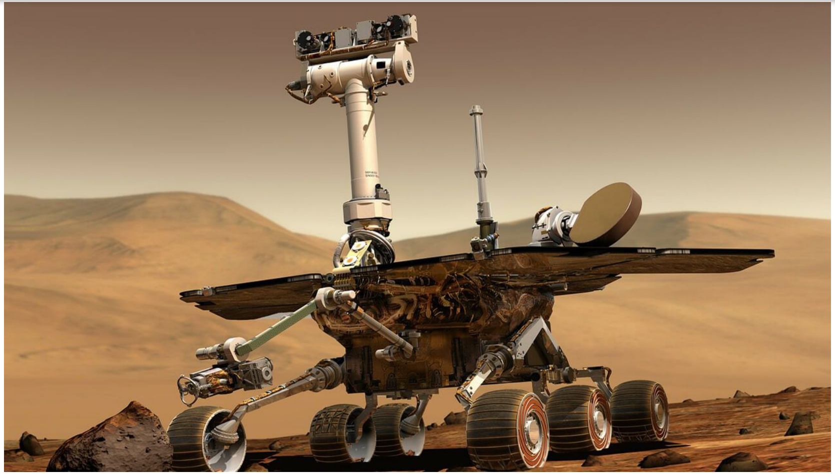
Wheeled Robots



⇒ **Robots with wheels!**

- Ball robots, 2/3/4-wheeled, or omni-wheeled
- Low-power and simple motion kinematics
  - Mainly ground (2D) robots
  - 2-DOF/3-DOF locomotion
- Application: warehouse, agriculture, and food delivery robots

Also Wheeled Robots!



⇒ Advanced AI features

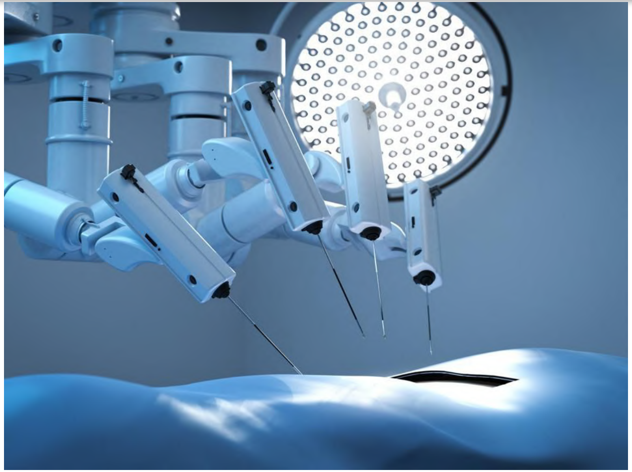
- Remote operation, long-term operation
- On-board AI computing
- Mainly ground (2D) robots
- <u>Example:</u> autonomous driving
- <u>Future:</u> self-driving cars!

Robot Manipulators

⇒ **Manipulators**

- Very specific ‘automated’ operation
- Closed-form motion kinematics
- Fast, efficient, low-power, and deterministic
- Vast mature state-of-the-art literature
- <u>Application:</u> Mainly in industry assembly lines
- <u>Future:</u> robot learning from demonstration

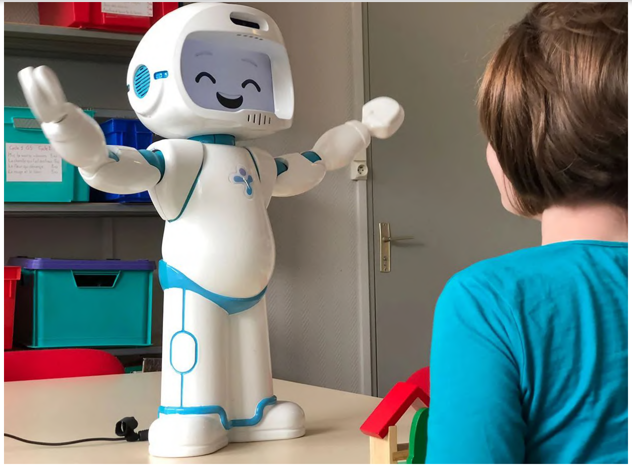
Surgical Robots

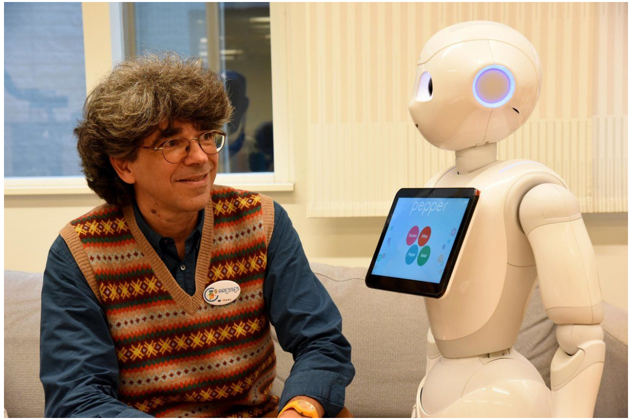


⇒ **Surgical robots and robotic instruments**

- Allows precise operation
- Shared control interface
  - Mostly remotely controlled, *ie,* tele-operation
  - With some autonomous features
  - Predefined and specific motion control
  - Usually closed-form kinematics
- <u>Application:</u> robotic-assisted surgery
- <u>Future:</u>
  - AR/VR capabilities for distant operations
  - Fully autonomous surgeons

Assistive And Social Robots



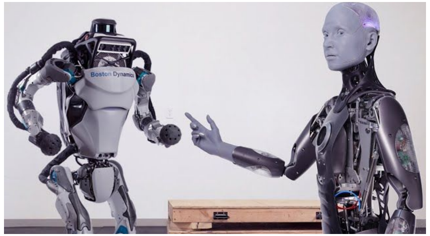


⇒ Assistive and social robots

- Social and interactive features are important
  - *Socially compliant* motion and gaze
  - Application-specific interaction
- Vast literature in human-robot interaction
- Application: assistive living for elderly, educational and guide industry

Legged And Humanoid Robots







⇒ Legged, bipeds, and humanoid robots

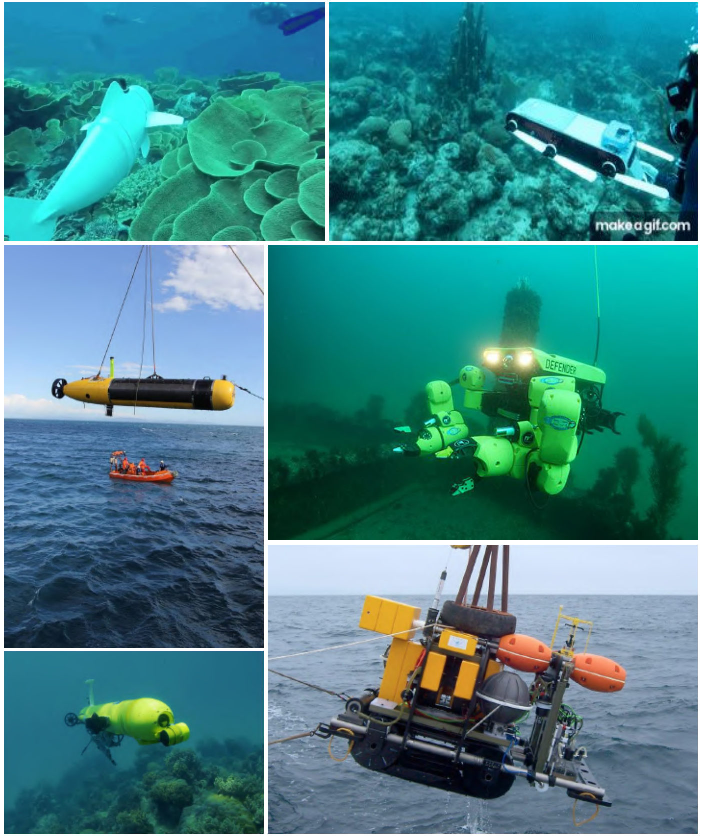
- Inspired by biological systems
  - Motion kinematics like humans, dogs, spiders!
- Primary research literature on
  - Stability control and motion control
  - Human-robot *cooperative missions*
- Have a history of being the forefronts of AI
- <u>Application:</u> human assistance in workplace, defense
- <u>Future</u>: more social/useful dog robots, pet robots, advanced warfare, planetary missions, *sentience!*

Aerial Robots

⇒ UAVs (Unmanned Aerial Vehicles)

- Complex 6-DOF motion kinematics
  - Multi-rotor, fixed-wing, single-rotor, etc.
- On-board *hard real-time* operation
- Vast literature and industrial advancements
- <u>Application:</u> agriculture, archeology, defense, mapping, and entertainment industry
- <u>Future:</u> food/package delivery, long-term flights

Underwater Robots



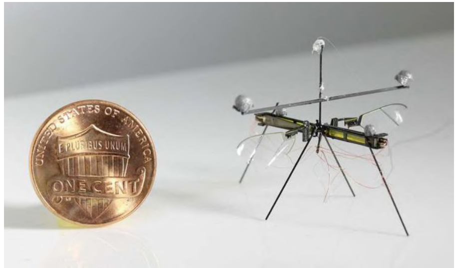
⇒ **AUVs (Autonomous Underwater Vehicles)**

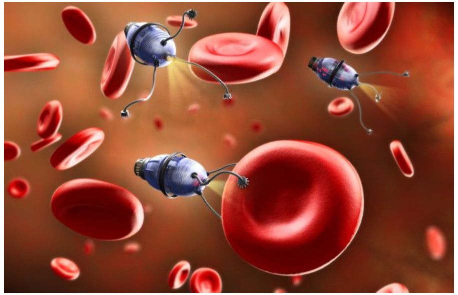
- 6-DOF or 3-DOF motion kinematics
  - Flipper or thruster based motion
  - Biological motion kinematics
- On-board *real-time constraints*
  - No GPS and WiFi/cloud connectivity
- Defense and industrial interests

- Application: marine archeology, biology, subsea monitoring, inspection, mapping, and exploration
- Future: full autonomous operation, 3D sensing, long-term monitoring, underwater positioning systems

- Other types:
  - **ROVs:** Remotely Operated Vehicles (*ie,* drones)
  - **ASVs:** Autonomous Surface Vehicles

Micro-Robots And Soft Robotics



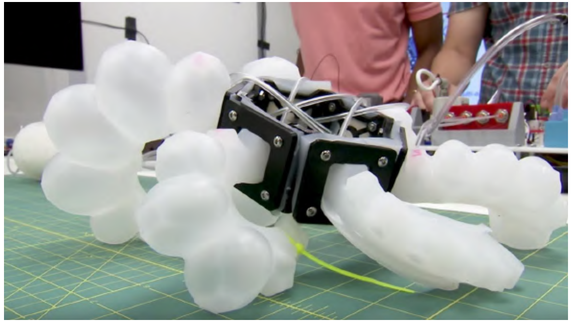


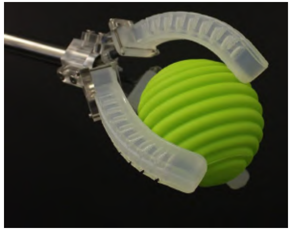
⇒ Microbots

- *Ultra low-power* miniature robots
  - Single microcontroller system
  - Specialized submicron motion
- Application: defense and medical robots
- Future: autonomous in-vivo missions

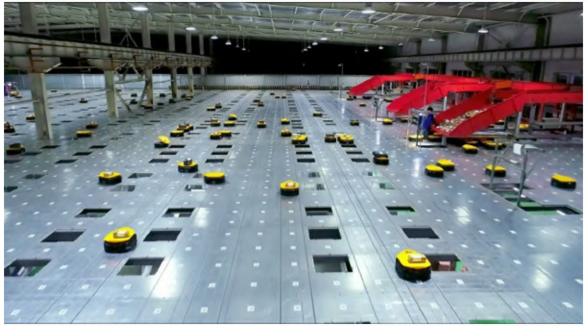
⇒ Soft robotics

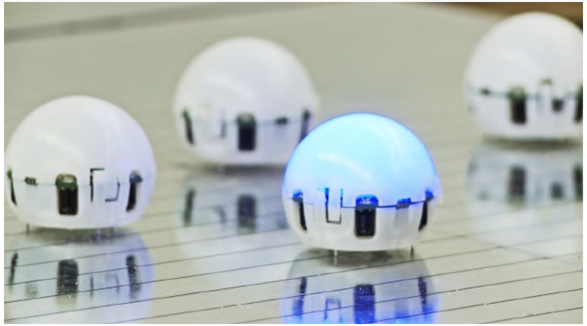
- Compliant materials instead of rigid links
- Unique design, control, and fabrication
  - Specialized submicron motion
  - Physically flexible body parts
- *Soft actuation & motion control* systems
- Application: surgical, wearable, and wearable and rehabilitation robots
- Future: smart exoskeletons





Swarm Robotics





⇒ **Swarm of robots!**

- A team of robots performing a coordinated task
  - Known as the *swarm intelligence*
  - Bio-inspired concept (from bees, ants, birds, etc.)
- Two main features: *communication* and *coordination*
- Vast literature in collective swarm behavior
- Application: packaging industry, agriculture, micro-robotics, and entertainment industry (drone shows)
- Future: defense with multi-agent missions, search-and-rescue

Brainstorm!

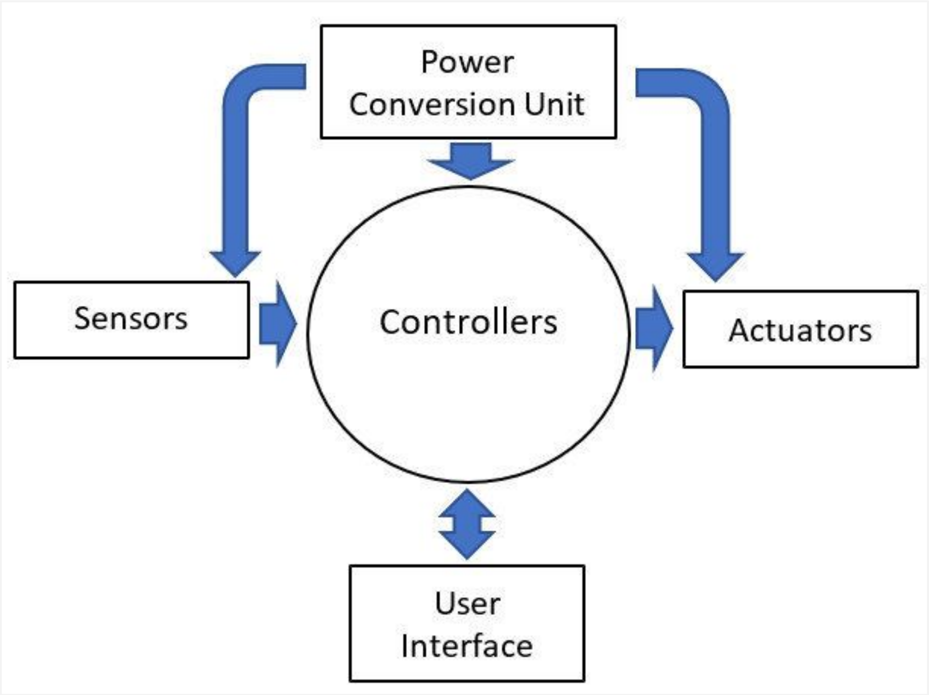
⇒ **Give real-world examples…**

- A multi-robot collaborative system
  - Mobile
  - Stationary

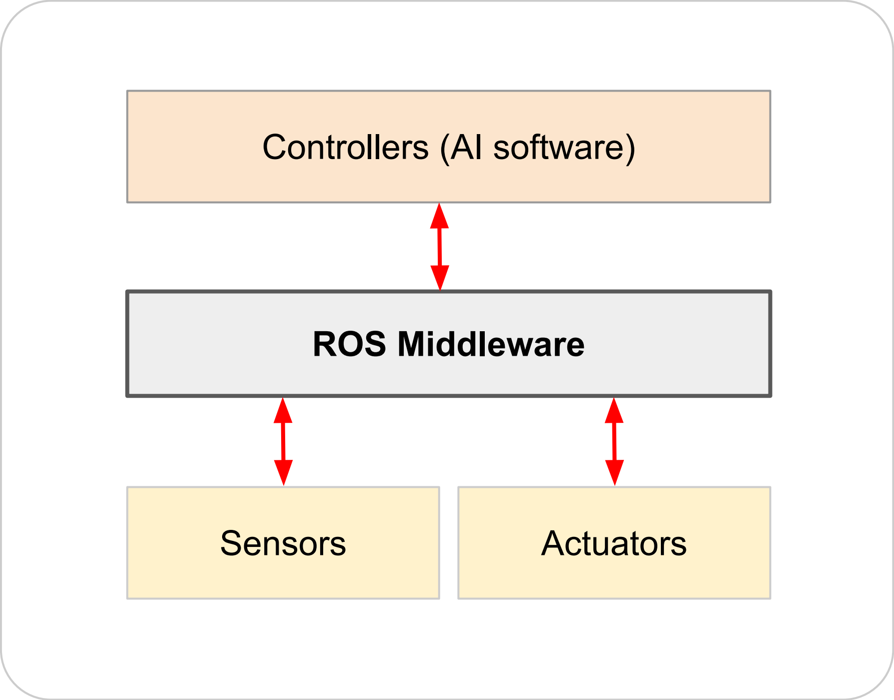
- A human-robot team in
  - Cooperative missions
  - Non-cooperative missions

- Single-agent interactive robot
  - Social settings
  - Industrial / commercial settings

Components Of A Robotic System

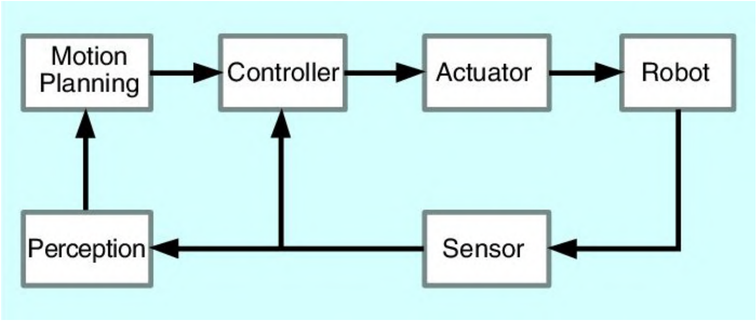


Basic Components

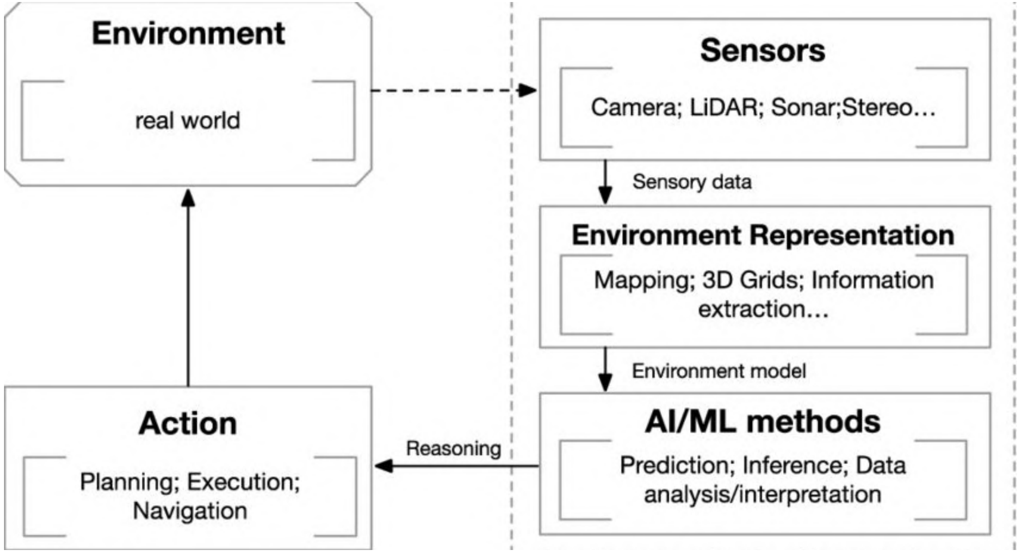


The 'ROS' Design

Computational Pipeline

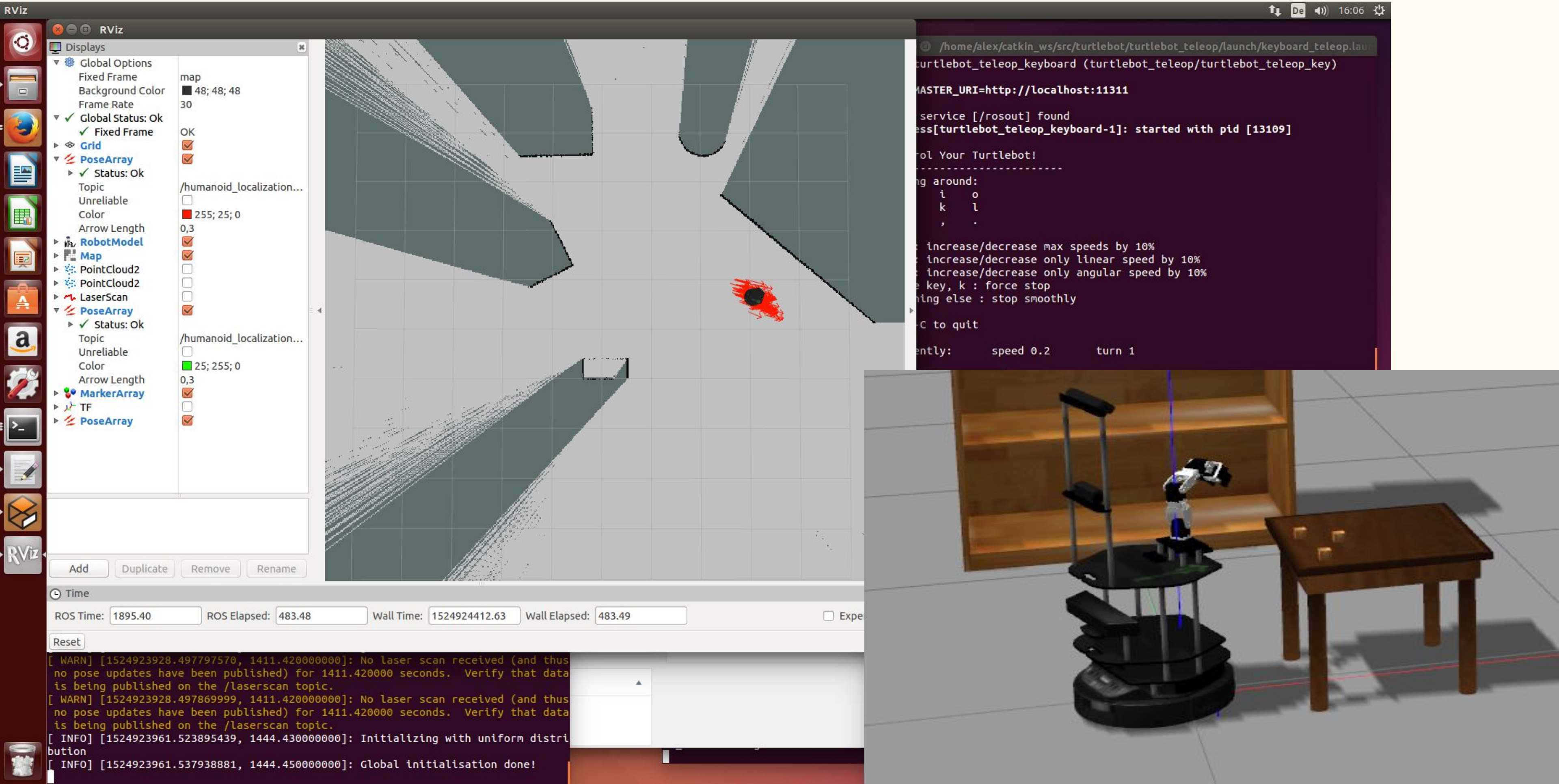


Perception-Planning-Control Loop



Agent-Action Perspective

ROS: Robot Operating System



What is ROS?

⇒ A middleware "OS" for robotics

- Open source *software packages*
    - Components + Tools + Interfaces
- For general-purpose *robot programming + hw/sw interfacing*
    - Actuators: things that move
    - Sensors: things that read the world
    - Control system: robots brain (AI functions!)
- Works best with linux distributions
- Visit ros.org for an introduction

ROS.org

⇒ Important concepts

- ROS nodes, services, topics, packages
- Data input (topic subscription) and output (topic publish)
- Data saving (ros bagging) and playing
- ROS simulators and learning packages
- *Moving/controlling a robot with ROS*

Pointers For Getting Started In ROS/ROS2

⇒ **Install ROS *melodic* or *noetic* (ROS 1)**

- Preferred: Linux laptops or Raspberry PI or Jetson Nanos
- Follow the instructions:
  - Getting started: https://www.ros.org/blog/getting-started/
  - Installation: https://wiki.ros.org/ROS/Installation
- Make sure to install the correct distribution for your platform

⇒ **ROS2 documentation:** https://docs.ros.org/

ROS.org

⇒ **Learn basic ROS functionalities**

- ROS Noetic tutorials by Robotics Back-End
- ROS Noetic tutorials by Emil Vidmark
- ROS2 Humble tutorials by Robotics Back-End
- Or browse any other resources!

All The Best