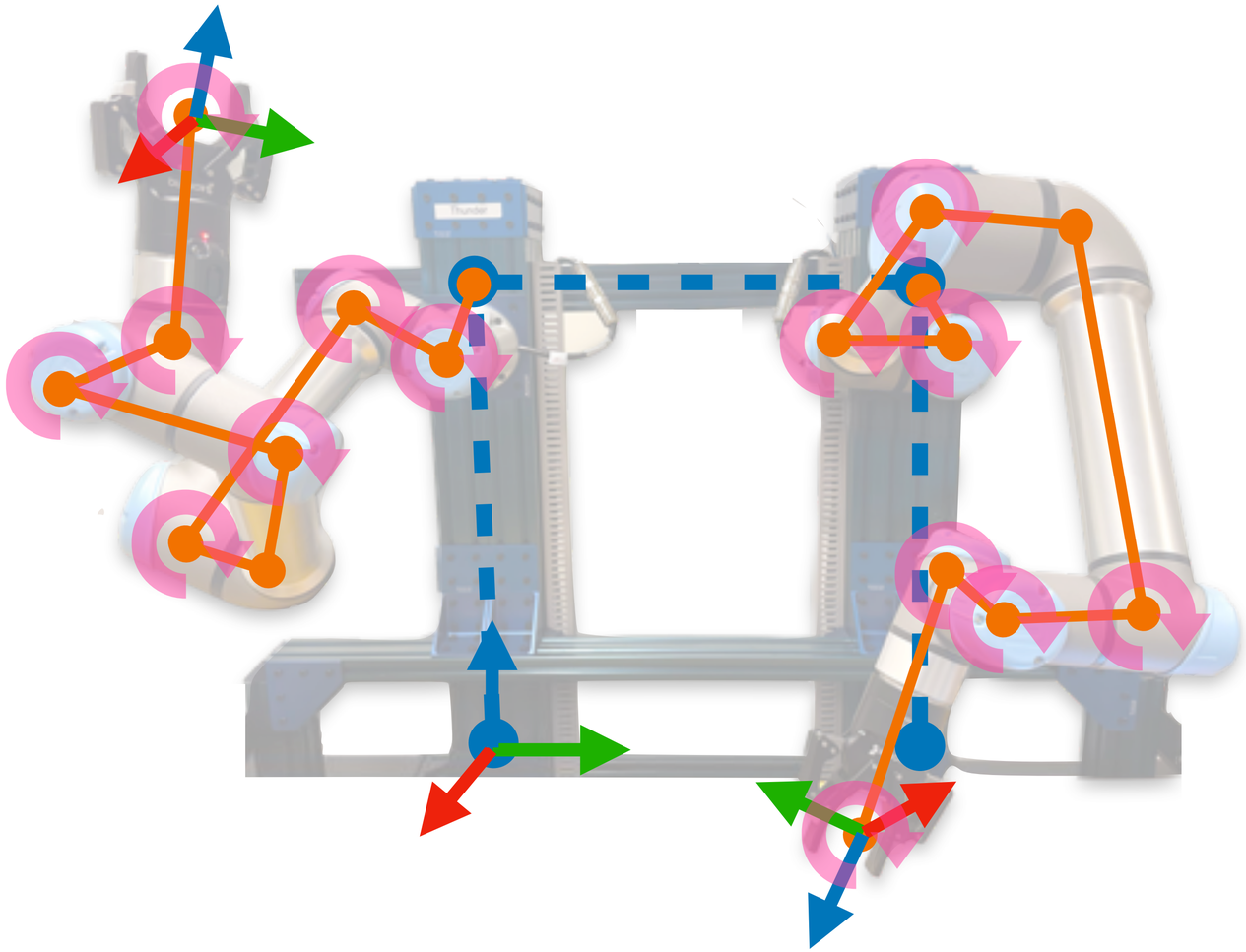
Introduction to Intelligent Robotic Systems



CSCI 5551

Fall 2023 (Section 002)

University of Minnesota

Welcome to 5551!

Section - 002

Course Staff

Section - 002

- **Instructor: Karthik Desingh *(he/him)***
  - Assistant Professor, CS&E | MnRI
  - kdesingh@umn.edu
  - OH: Mondays 8:30-10:00 am CT Shepherd 2-234

- **TA: Chahyon Ku *(he/him)***
  - MS Student in Robotics
  - ku000045@umn.edu
  - OH: T, Th 10:00-11:00 am CT Keller Hall 2-209

Acknowledgement

- This course builds on and is indebted to materials from:
  - Prof. Chad Jenkins (University of Michigan) and the staff of autorob.org
  - Prof. Nikolaos Papanikolopoulos (University of Minnesota)
  - Prof. Junaed Sattar (University of Minnesota)

Section - 001
M 06:30 PM – 09:00 PM

What are intelligent robotic systems?

What are intelligent robotic systems?

"systems that provide intelligent services and information by interacting with their environment, including human beings, via the use of various sensors, actuators and human interfaces"

What are intelligent robotic systems?

"systems that provide intelligent services and information by interacting with their environment, including human beings, via the use of various sensors, actuators and human interfaces"

What are intelligent robotic systems?

It is getting very hard to define this term.
For the sake of this course,
let us call this **“ability to operate with some autonomy”**

What are intelligent robotic systems?

What are intelligent robotic systems?

HaptX: https://www.youtube.com/watch?v=uwYtwQtoOh0

What are intelligent robotic systems?

Fully Teleoperational

Fully Autonomous

Spectrum of Shared autonomy

HaptX: https://www.youtube.com/watch?v=uwYtwQtoOh0

TUM/IAS group: https://www.youtube.com/watch?v=cTCJSNjTdo0

What are intelligent robotic systems?

…systems that can perform Sense-Plan-Act….

What are intelligent robotic systems?

…systems that can perform Sense-Plan-Act….

Zhiqiang Sui et al. 2017

What are intelligent robotic systems?

…systems that can perform Sense-Plan-Act….

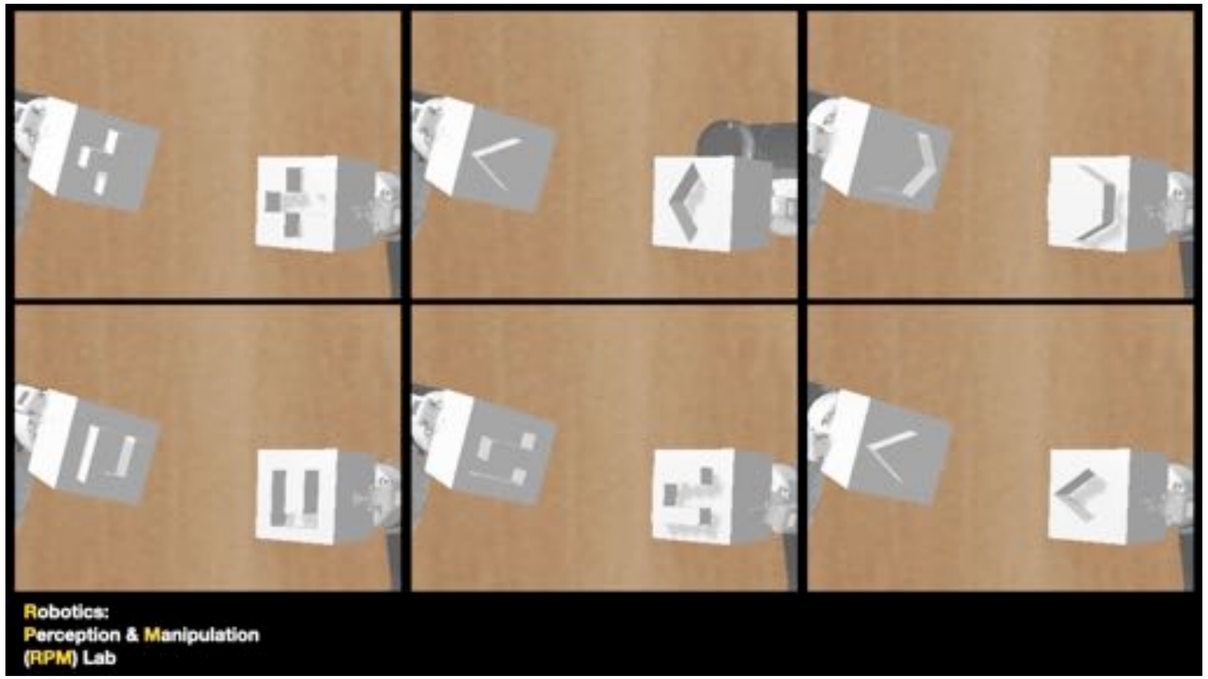
Zhiqiang Sui et al. 2017

What are intelligent robotic systems?

…systems that can perform Sense-Plan-Act….

… can also learn skills … transfer these skills … adapt to new environments …

Carl Winge et al. 2022



Chahyon Ku et al. 2023

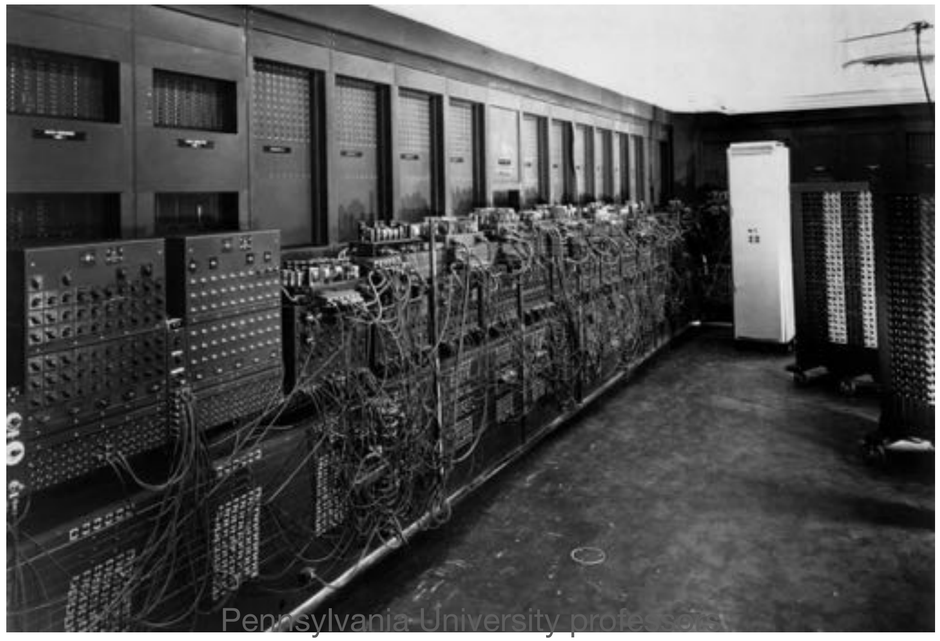
What are intelligent robotic systems?

…systems that can perform Sense-Plan-Act….

… can also learn skills … transfer these skills … adapt to new environments …

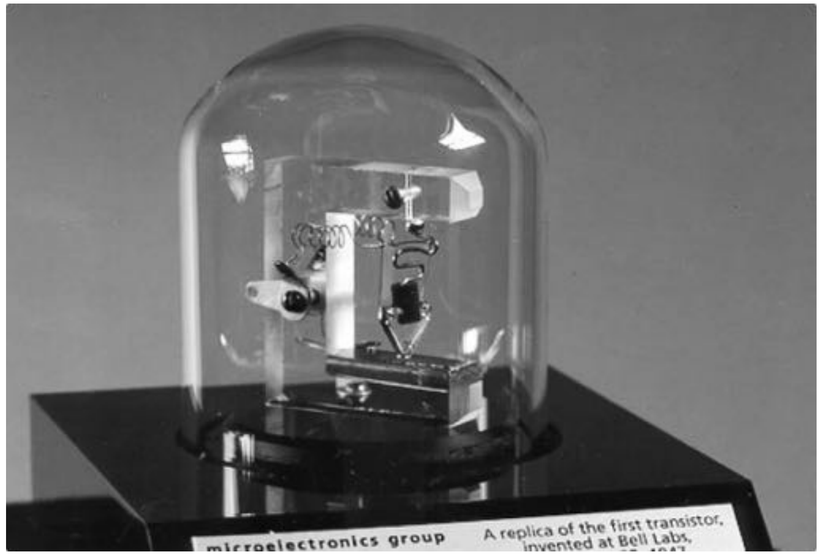
Bahaa Aldeeb et al. 2023

History of Computers, AI and Robotics



1944

John Mauchly and J. Presper Eckert build the ‘grandfather’ of digital computers, the Electronic Numerical Integrator and Calculator (ENIAC)



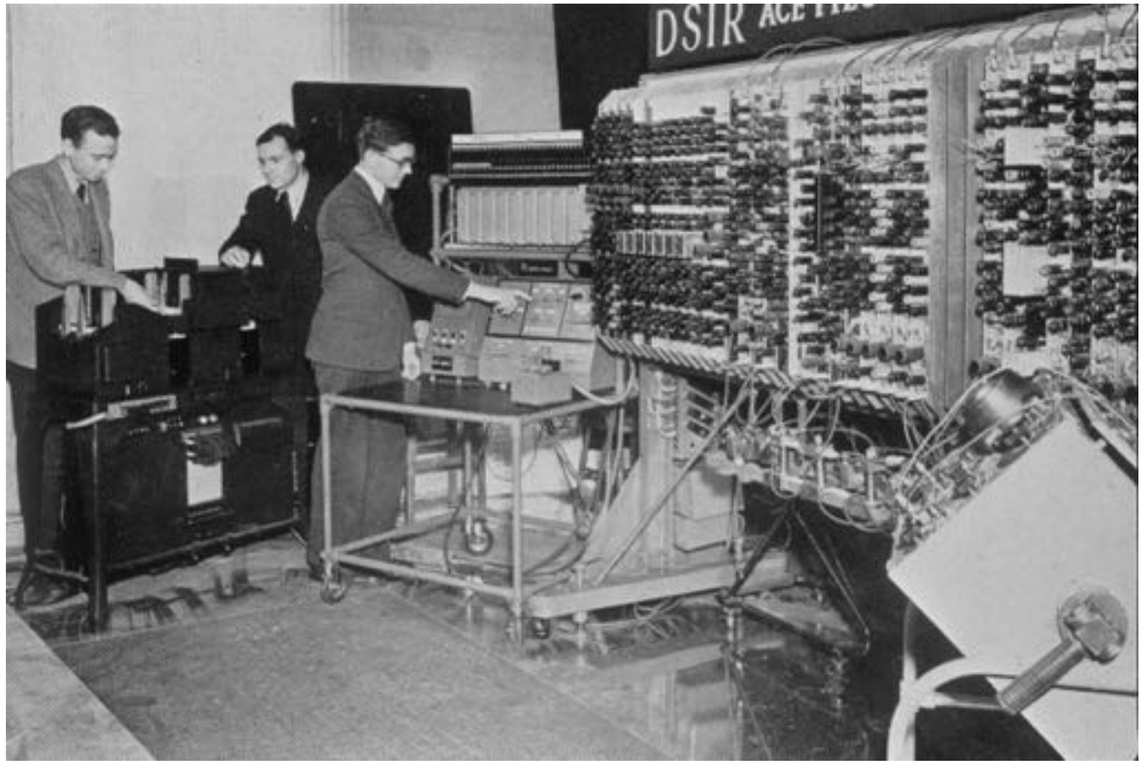
1948

Researchers William Shockley, John Bardeen and Walter Brattain at Bell Laboratories invent the transistor.

1950

‘I,Robot’ by Issac Asimov is published, laying the foundations for the idea of robots in culture.

https://everydayrobots.com/



Alan Turing introduces ‘The Turing Test’ —
a test of a machine's ability to exhibit intelligent behavior
equivalent to, or indistinguishable from, that of a human.

1950

Grace Hopper develops COBOL, the first computer language.
The second, FORTRAN, is developed by a team of
IBM programmers a year later.

1953

https://hotcorn-cdn.s3.amazonaws.com/wp-content/uploads/sites/2/2020/09/22125112/bletchleypark-pilotace-scaled.jpg
https://everydayrobots.com/

Dartmouth conference coins the term
'artificial intelligence' and
launches the field of AI.
IBM mainframes are used in early experiments

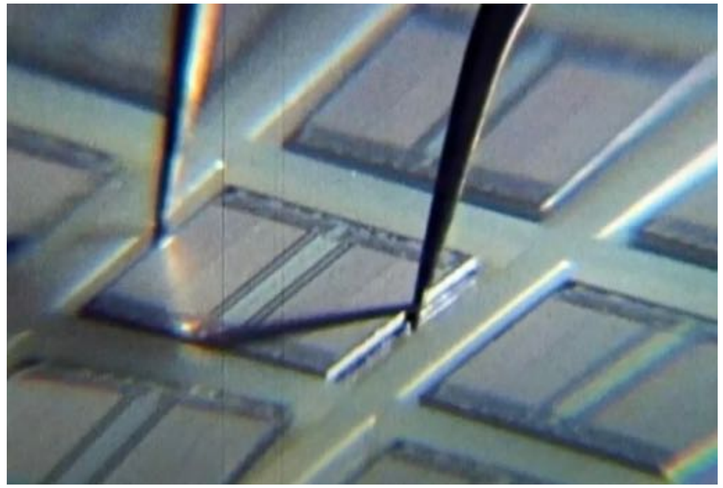
Five of the attendees of the 1956 Dartmouth Summer Research Project on Artificial Intelligence reunited at the July AI@50 conference. From left: Trenchard More, John McCarthy, Marvin Minsky, Oliver Selfridge, and Ray Solomonoff. (Photo by Joseph Mehling '69)

1956

IBM's chairman and CEO, Thomas J. Watson Jr.,
bets the company's future on the IBM Series/360 —
the largest privately-financed commercial project in history.
The risk pays off, changing the computer industry forever.
Work is revolutionized, productivity is enhanced and
countless new tasks become possible.

1964

https://everydayrobots.com/



Intel and Ted Hoff introduce the first microprocessor, the Intel 4004.

Intel co-founder, Gordon Moore, theorizes that computing would dramatically increase in power, and decrease in relative cost, at an exponential pace. The insight, known as Moore's Law, becomes the golden rule for the electronics industry, and a springboard for innovation.

1971

Steve Wozniak and Steve Jobs release the Apple 1

1976

A year later, Apple releases the Apple II

1977

https://image.cnbcfm.com/api/v1/image/100932798-128279719-1.jpg?v=1583960525

https://everydayrobots.com/

The U.S. Defense Department funds the first experimental computer network— ARPANET. It connects computers everywhere, and is a forerunner to the internet.

1966

```
<!DOCTYPE html>
<html>
<body>

<h1>What will people do now we have the internet?</h1>

<p>Surprisingly, it turns out that we really like looking at pictures of cats.</p>
```

Tim Berners-Lee and his colleagues at CERN develop hypertext markup language (HTML) and the uniform resource locator (URL), giving birth to the first incarnation of the World Wide Web.

1995

IBM's Deep Blue Computer, a form of AI, beats reigning world chess champion Gary Kasparov.

1997

https://i.insider.com/55947fbf2acae7b7188b5388?width=750&format=jpeg&auto=webp
https://everydayrobots.com/

Larry Page & Sergey Brin, two computer science graduate students from Stanford University, pioneer a new way to search for and find information on the web. They call their invention ‘Google’.

1998

Four founders start a company called ‘Android

2003

https://i.insider.com/55947fbf2acae7b7188b5388?width=750&format=jpeg&auto=webp
https://indonesiamendesain.com/wp-content/uploads/2020/06/original-google-logo-font.jpg
https://everydayrobots.com/

Steve Jobs unveils the iPhone at Macworld

Search becomes intuitive.
Maps are intelligent.
Work is more productive than ever

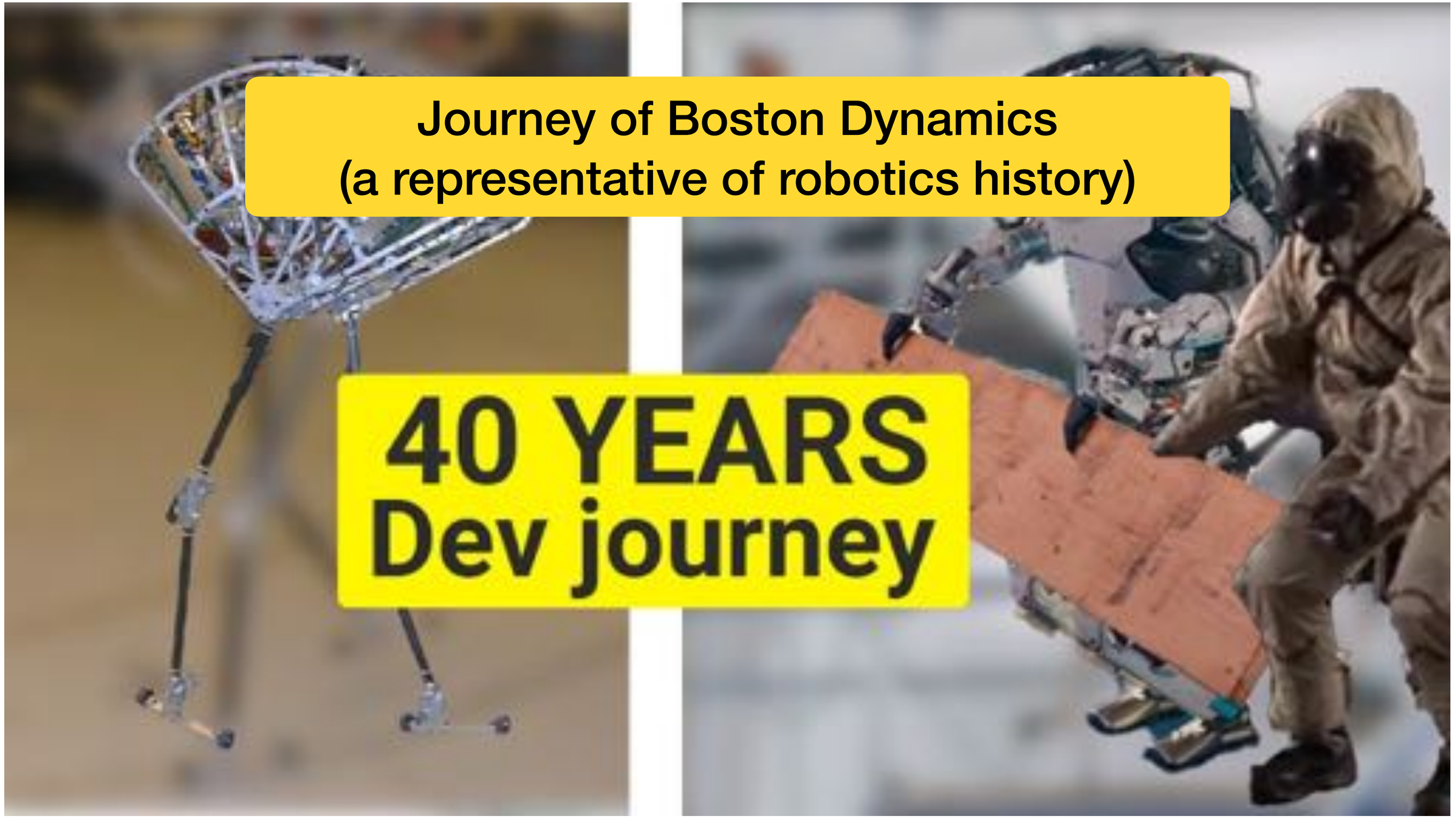
2007

https://i.insider.com/587374fadd0895e1148b47e7?width=1136&format=jpeg

https://everydayrobots.com/

So, this is computing and AI…
What was happening in robotics?

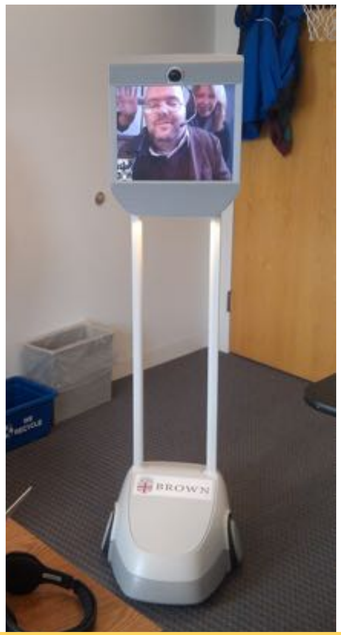
Journey of Boston Dynamics
(a representative of robotics history)

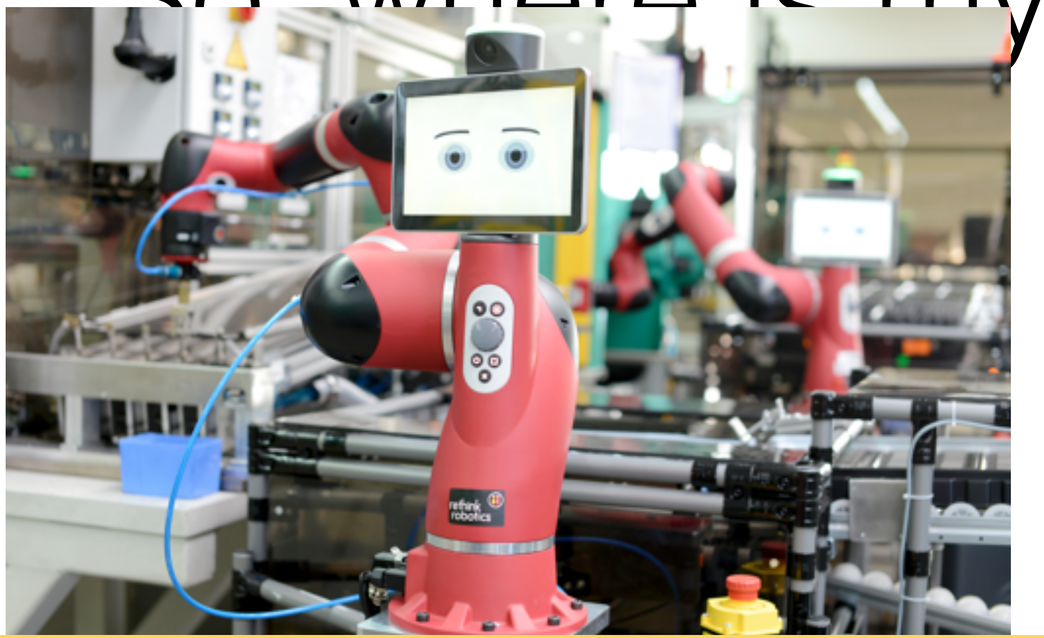


https://youtu.be/qtos3ezujn4?feature=shared

So, where is my robot?

So, where is my robot?

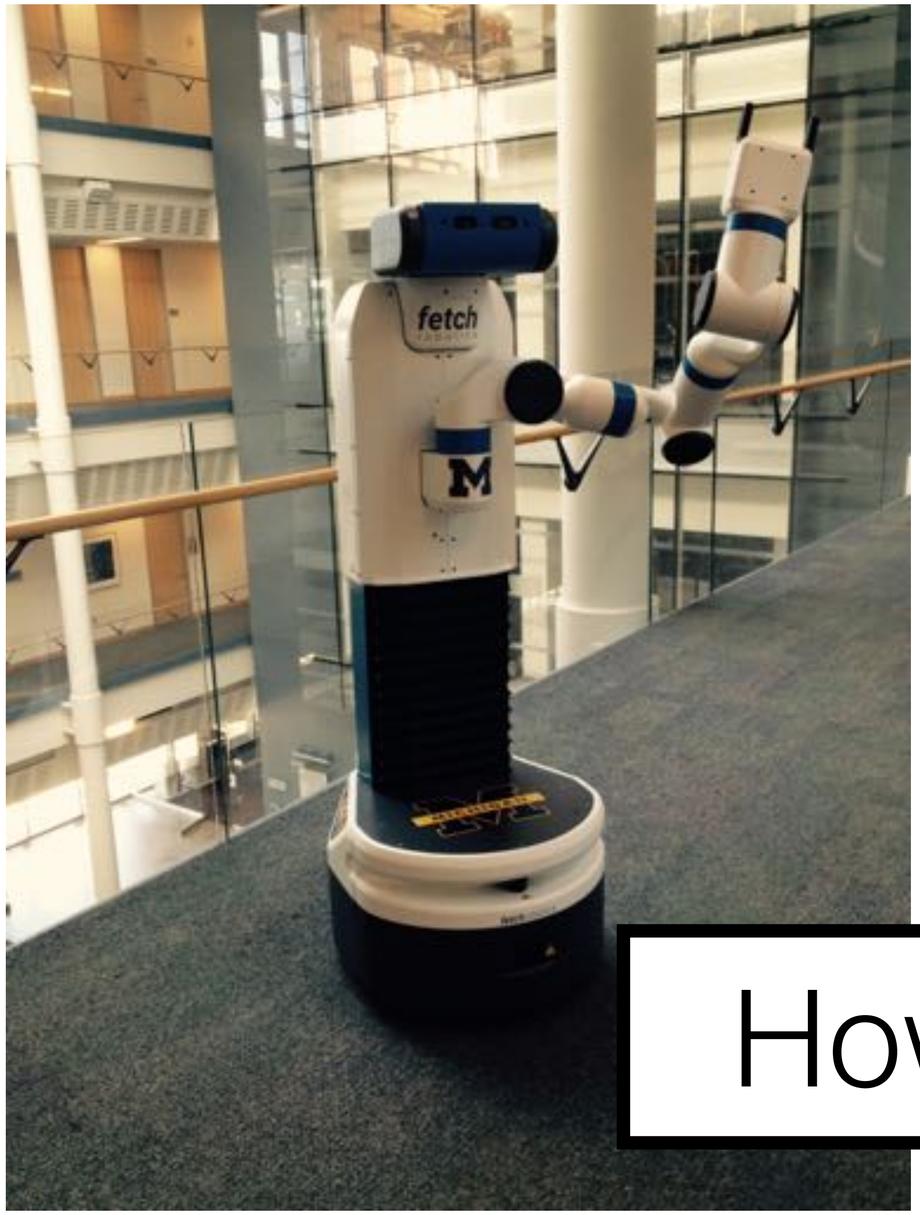


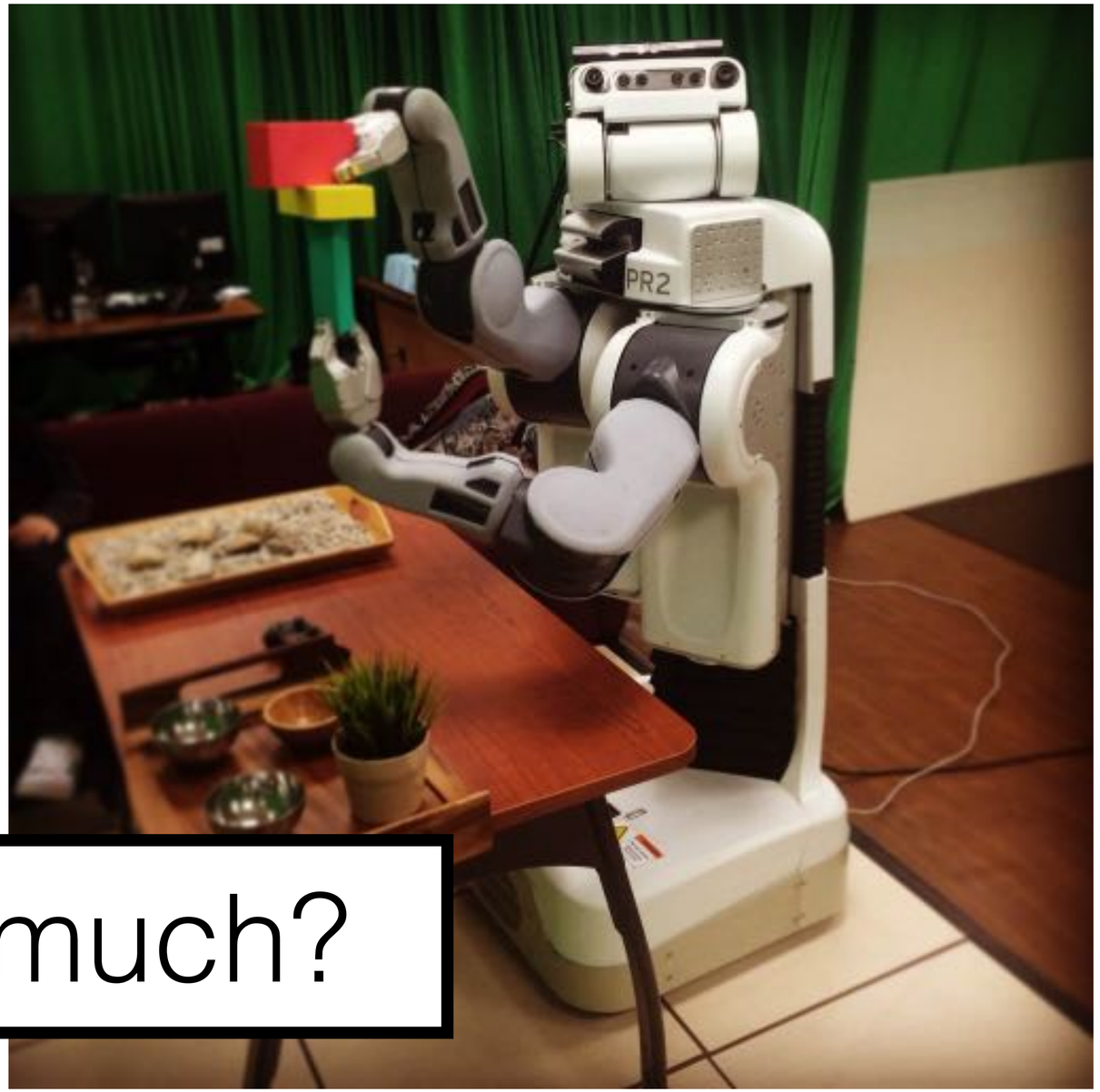


Slide borrowed from Michigan Robotics autorob.org

Mobile Manipulation Robots

Slide borrowed from Michigan Robotics autorob.org





How much?

Slide borrowed from Michigan Robotics autorob.org

Cost

$400K

$100K

Willow Garage PR2

Fetch

2009

2015

Time

2002

Willow Garage PR2

Fetch

$400K

$100K

2009

2015

Time

Cost

NASA Robonaut

Linear algebra enables us to fit a model to data

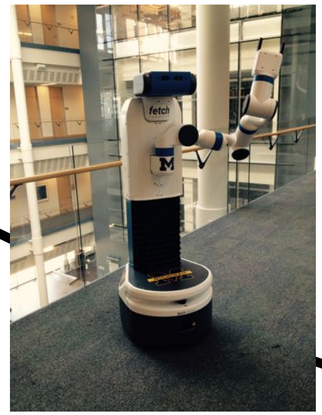
(polynomial regression in this case)

$1.5M

Willow Garage PR2

$400K

Fetch



$100K

2001

2009

2015

Time

Cost

NASA Robonaut

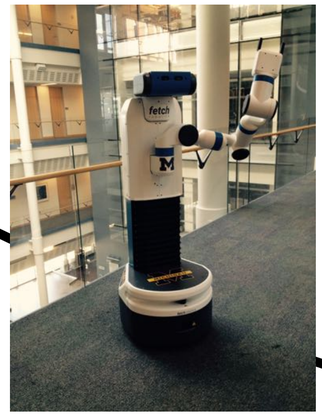
Linear algebra enables us to fit a model to data

(polynomial regression in this case)

... and make predictions

Willow Garage PR2

Fetch



Your robot

$1.5M

$400K

$100K

$40K

2001

2009

2015

2023

Time

Cost

$1.5M

$400K

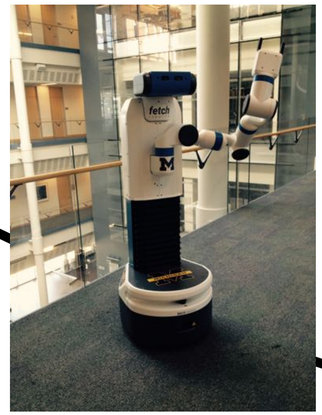
$100K

$40K

NASA Robonaut

Willow Garage PR2

Fetch



Your robot

2001

2009

2015

2023

Time

Cost

$1.5M

$400K

$100K

$40K

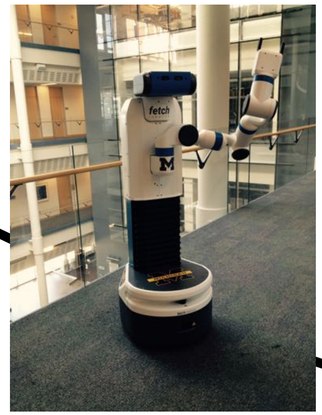
NASA Robonaut

Teleoperation

(“Remote Control”)

Willow Garage PR2

Fetch



Your robot

2001

2009

2015

2023

Time

Cost

$1.5M

$400K

$100K

$40K

NASA Robonaut

Teleoperation

(“Remote Control”)

2001

2009

2015

2023

Time

Cost

$1.5M

$400K

$100K

$40K

NASA Robonaut

Teleoperation

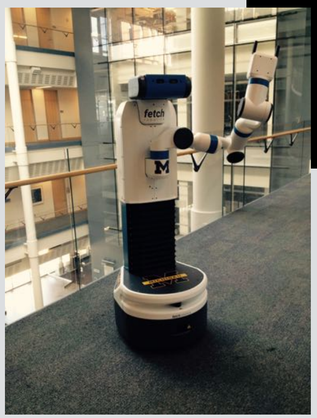
(“Remote Control”)

Pick-and-Place

(“Put that there”)

Willow Garage PR2

Fetch



2001

2009

2015

2023

Time

Slide borrowed from Michigan Robotics autorob.org

Cost

$1.5M

$400K

$100K

$40K

NASA Robonaut

Teleoperation

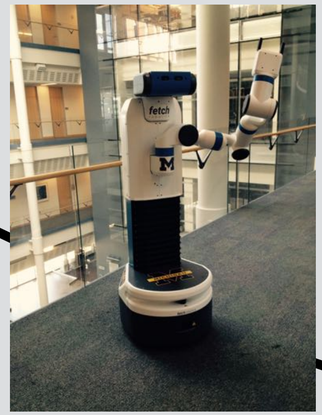
("Remote Control")

Pick-and-Place

("Put that there")

Willow Garage PR2

Fetch



Taskable autonomy

("Do this task for me")

Your robot

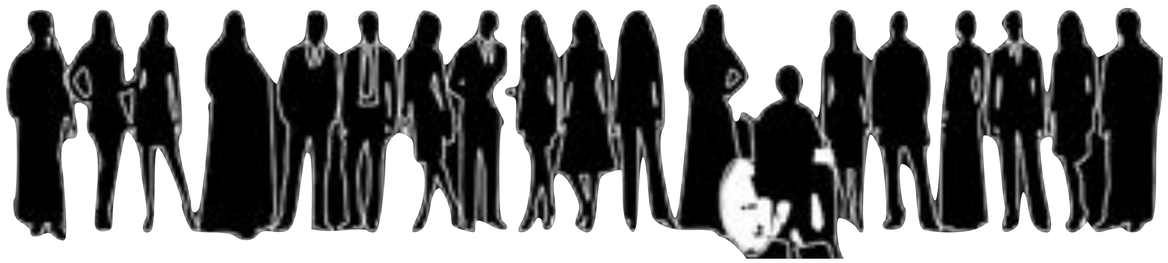
2001

2009

2015

2023

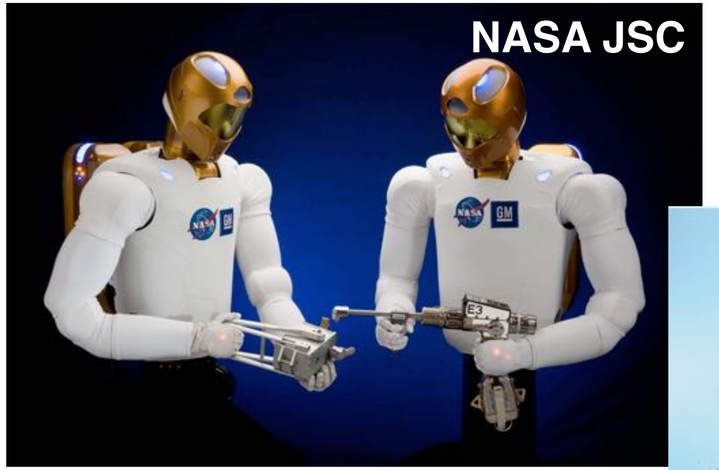
Time



Taskable autonomy

Pick-and-Place

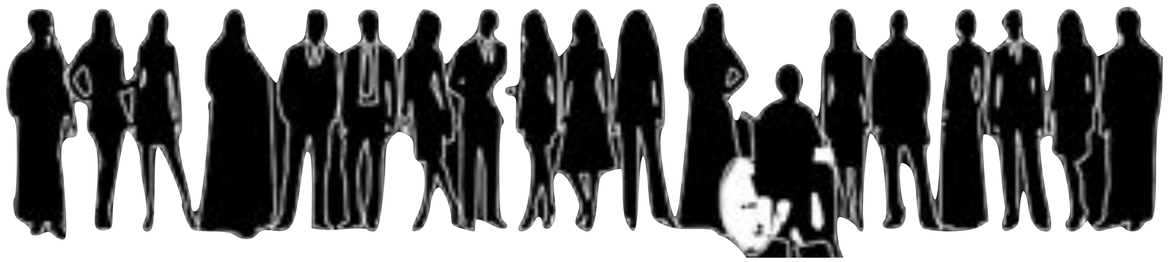
Teleoperation



TU Munich

Dexterous Manipulation

Harvard/Wyss

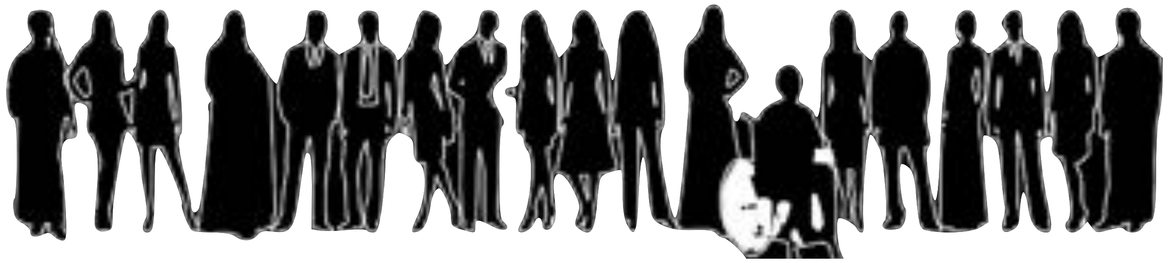


Taskable autonomy

Dexterous Manipulation

Teleoperation

Slide borrowed from Michigan Robotics autorob.org



Taskable autonomy

Dexterous Manipulation

Teleoperation

Operating system

From Wikipedia, the free encyclopedia

An **operating system** (**OS**) is system software that manages computer hardware, software resources, and provides common services for computer programs.

Time-sharing operating systems schedule tasks for efficient use of the system and may also include accounting software for cost allocation of processor time, mass storage, printing, and other resources.

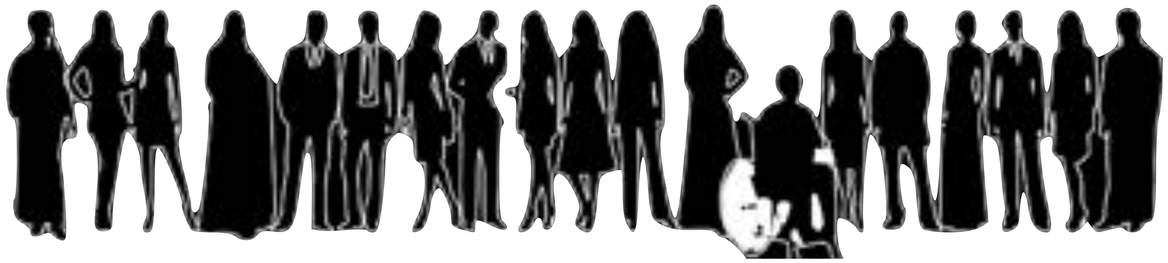
For hardware functions such as input and output and memory allocation, the operating system acts as an intermediary between programs and the computer hardware,[1][2] although the application code is usually executed directly by the hardware and frequently makes system calls to an OS function or is interrupted by it. Operating systems are found on many devices that contain a computer – from cellular phones and video game consoles to web servers and supercomputers.

The dominant desktop operating system is Microsoft Windows with a market share of around 82.74%. macOS by Apple Inc. is in second place (13.23%), and the varieties of Linux are collectively in third place (1.57%).[3] In the mobile sector (including smartphones and tablets), Android's share is up to 70% in the year 2017.[4] According to third quarter 2016 data, Android's share on smartphones is dominant with 87.5 percent with also a growth rate of 10.3 percent per year, followed by Apple's iOS with 12.1 percent with per year decrease in market share of 5.2 percent, while other operating systems amount to just 0.3 percent.[5] Linux distributions are dominant in the server and supercomputing sectors. Other specialized classes of operating systems, such as embedded and real-time systems, exist for many applications.

Contents [hide]

1 Types of operating systems
1.1 Single-tasking and multi-tasking
1.2 Single- and multi-user
1.3 Distributed
1.4 Templated
1.5 Embedded

Slide borrowed from Michigan Robotics autorob.org



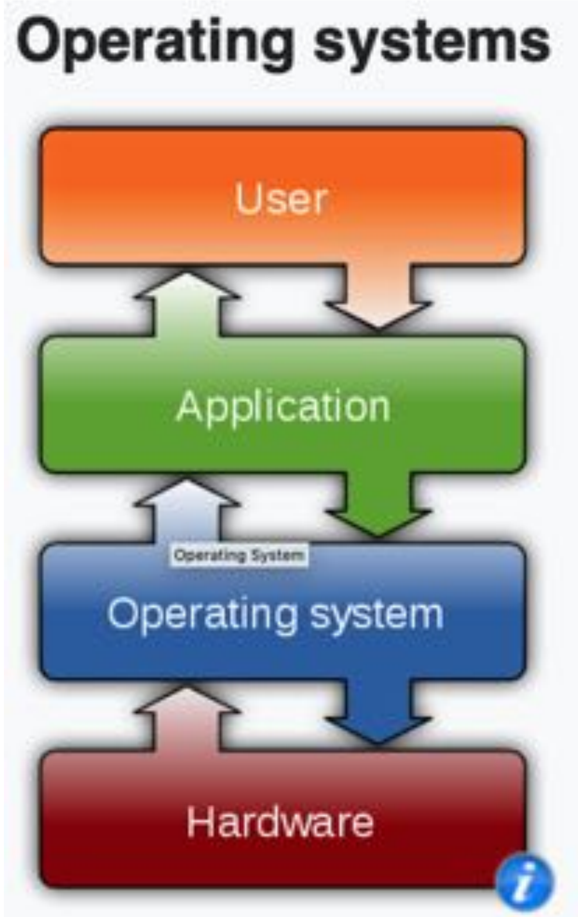
Taskable autonomy

Dexterous Manipulation

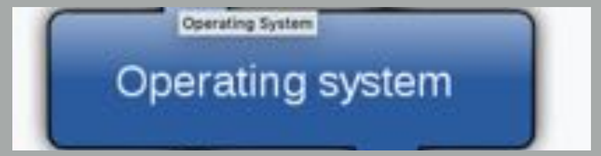
Teleoperation

An **operating system (OS)** is a special program that runs on the bare machine and hides the gory details of managing processes and devices.

- https://perldoc.perl.org/perlglossary.html#operating-system

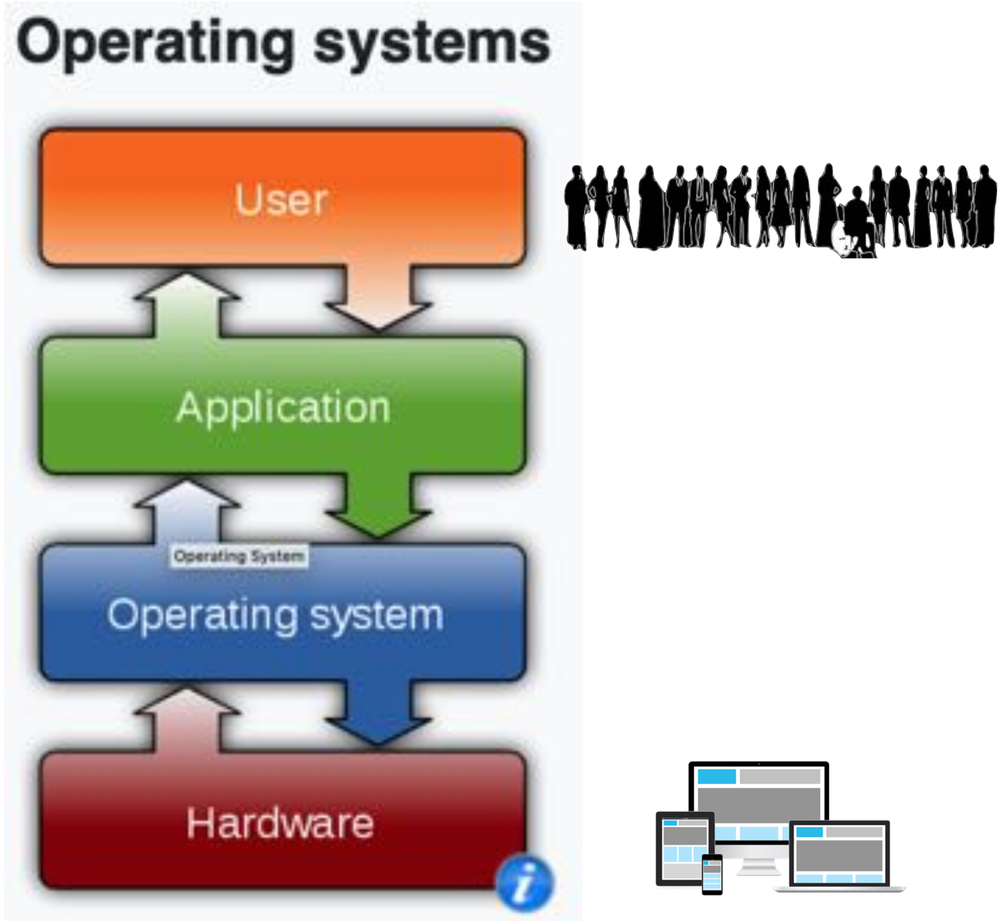


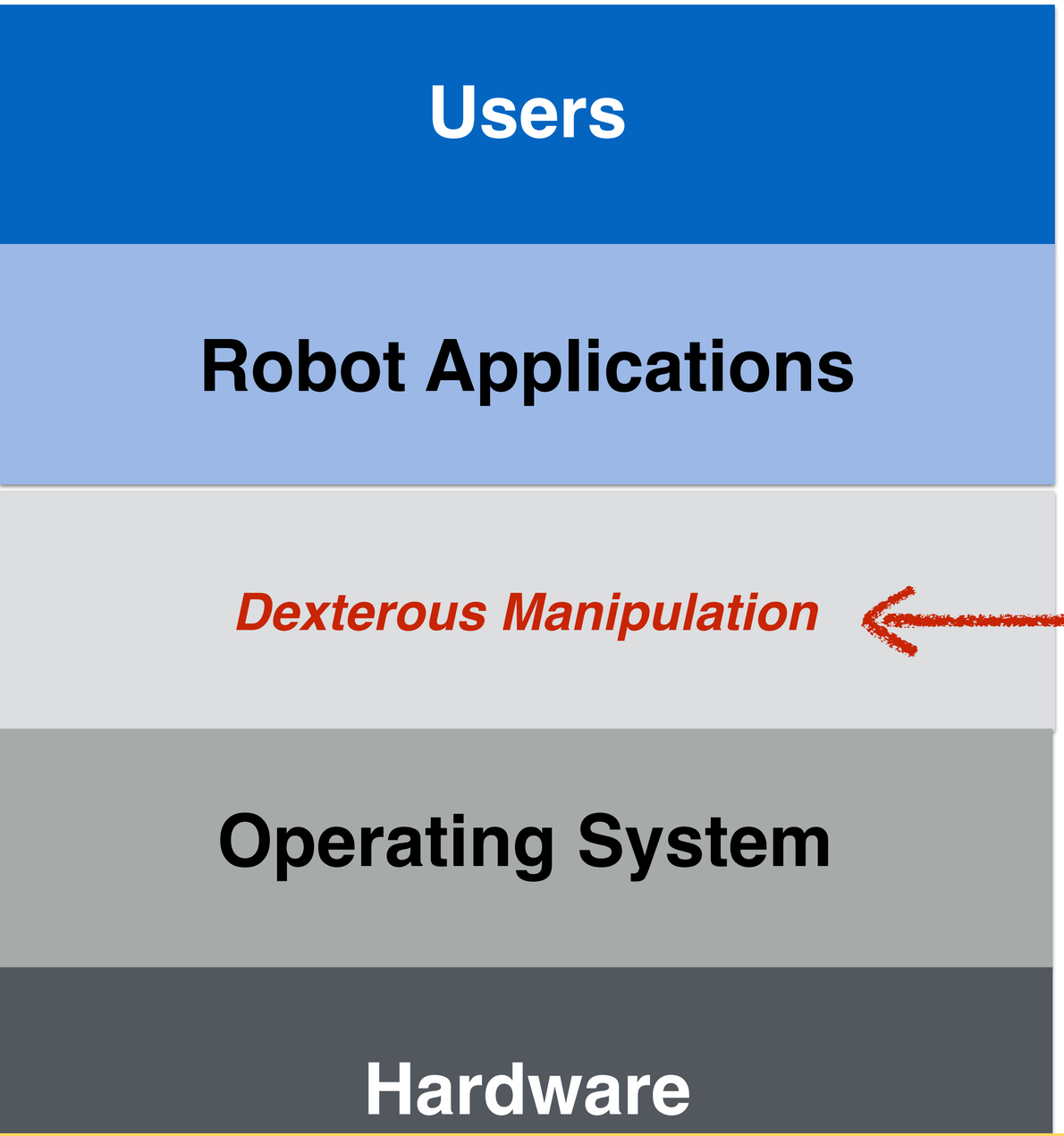
Dexterous Manipulation



An **operating system (OS)** is a special program that runs on the bare machine and hides the gory details of managing processes and devices.

- https://perldoc.perl.org/perlglossary.html#operating-system





Then, what is this?

Slide borrowed from Michigan Robotics autorob.org

Users

Robot Applications

Robot Operating System

Operating System

Hardware

A **robot operating system (robot OS)** is a special program that runs on the operating system and hides the gory details of controlling robot devices, autonomy processes, and sensorimotor routines.

This abstraction provides a platform for robot applications to run seamlessly across a wide variety of robots capable of mobility and/or dexterous manipulation.

Slide borrowed from Michigan Robotics autorob.org

Users

Robot Applications

Robot Operating System

Operating System

Hardware

Slide borrowed from Michigan Robotics autorob.org

Users

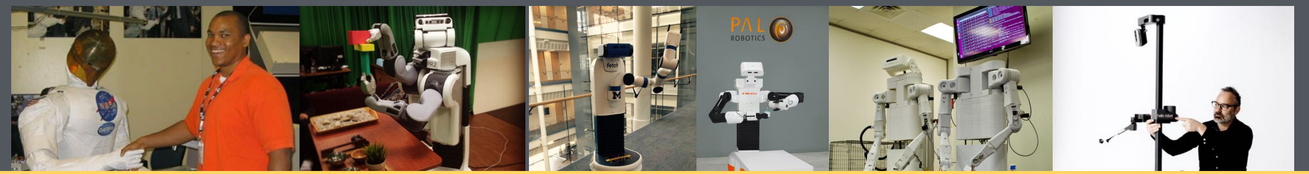
Robot Applications

Robot Operating System

Operating System




Hardware



Slide borrowed from Michigan Robotics autorob.org

Users

Robot Applications

Robot Operating System

Operating System

Hardware

Slide borrowed from Michigan Robotics autorob.org

Users

Robot Applications

Robot Operating System

LCM ROS Player YARP MOOS

Operating System

Hardware

Slide borrowed from Michigan Robotics autorob.org

Users

Robot Applications

Then, what is this?

Robot Operating System

LCM ROS Player YARP MOOS10

Operating System




Hardware

Slide borrowed from Michigan Robotics autorob.org

Users

Robot Applications

Apps of the Future…
"Do this task for me"

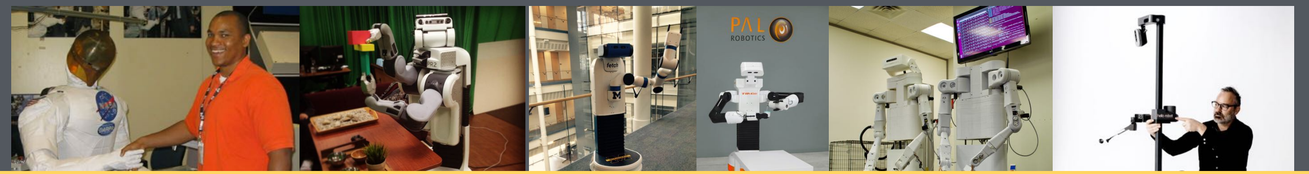
Robot Operating System

LCM ROS Player YARP MOOS10

Operating System




Hardware



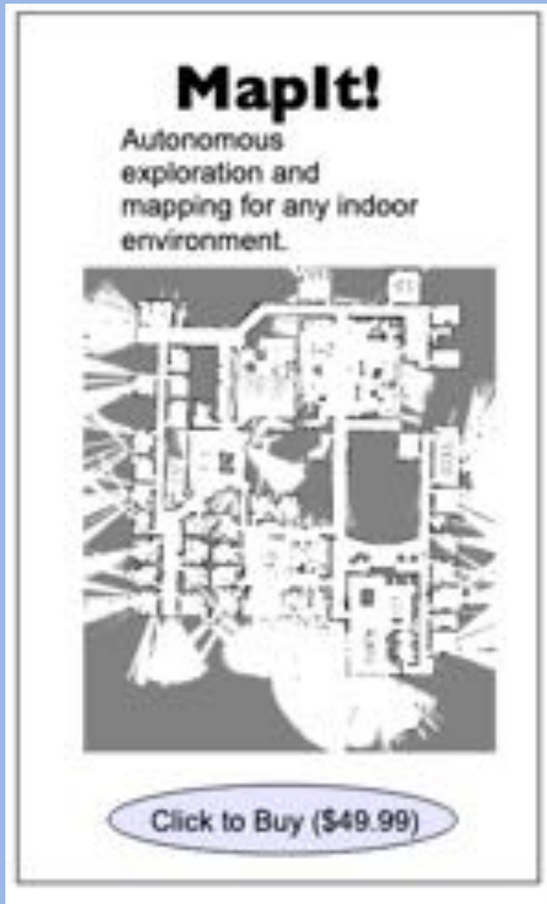
Slide borrowed from Michigan Robotics autorob.org

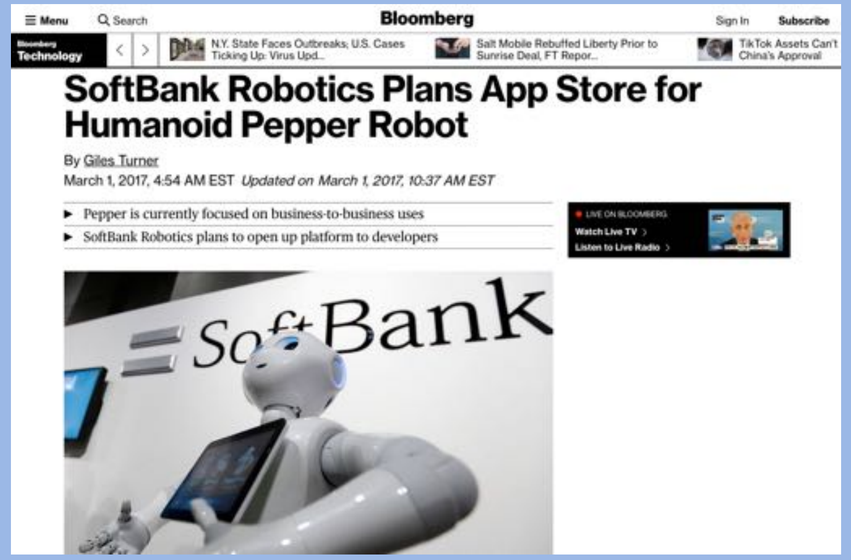
“Do this task for me”

Slide borrowed from Michigan Robotics autorob.org

Can we make your
world programmable ?

Slide borrowed from Michigan Robotics autorob.org





Can we make a **robot app store** ?

Slide borrowed from Michigan Robotics autorob.org

Willow Garage

What's a robot app?

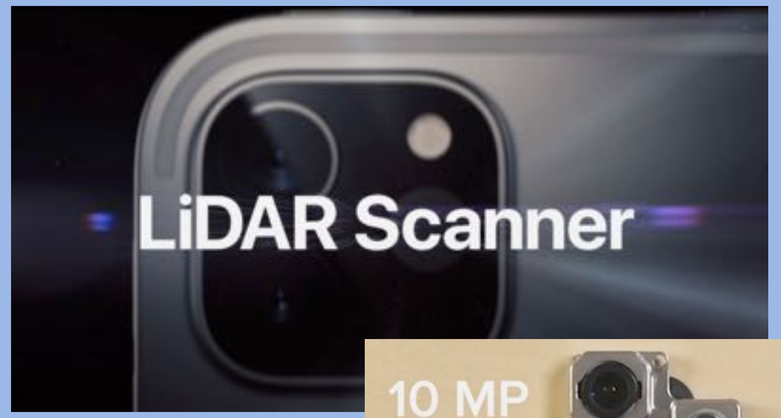
- In the near future
- Eventually:
  - CleanTheHouse
  - PatrolTheBuilding
  - ...
- For now:
  - demonstrations
  - experiments
  - challenge entries (!)

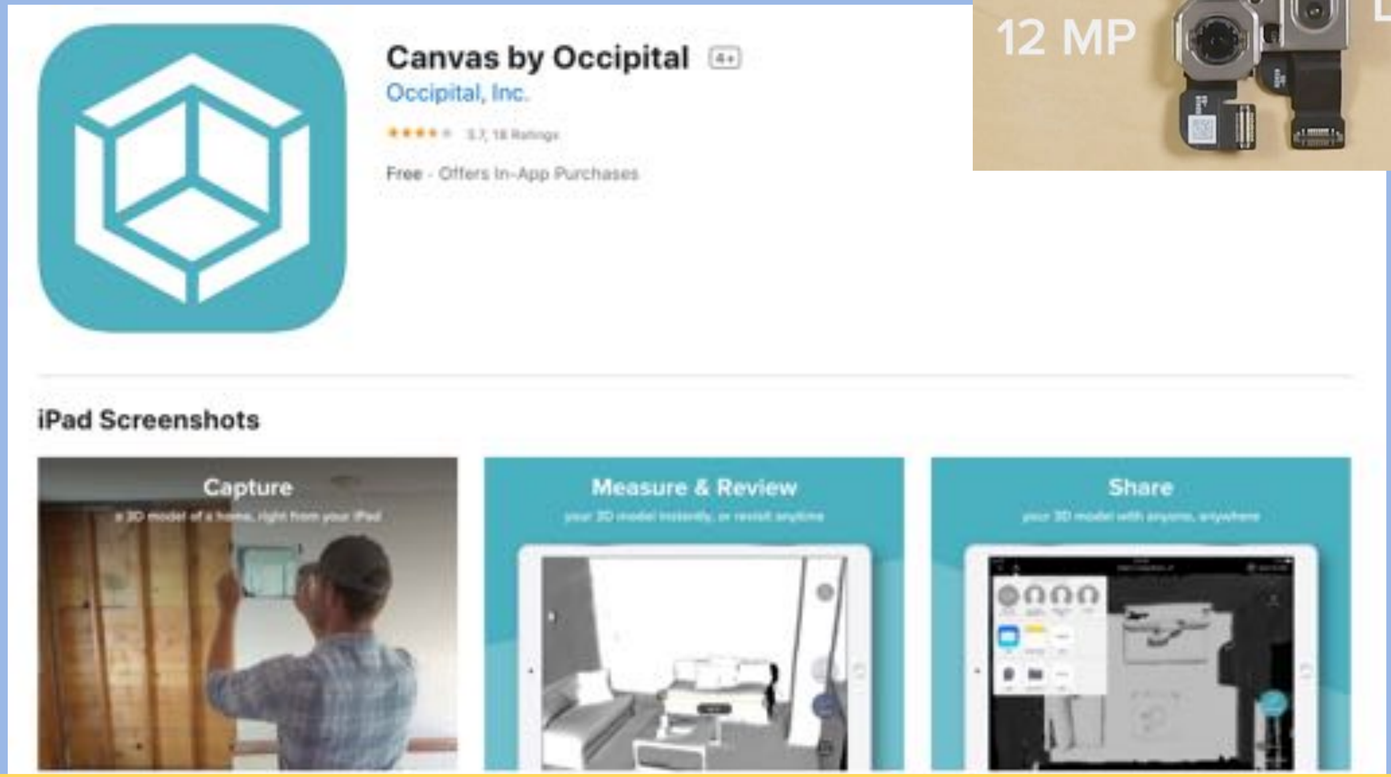
https://www.ros.org/presentations/2009-07_Gerkey_IJCAI-robotics-workshop.pdf

Slide borrowed from Michigan Robotics autorob.org

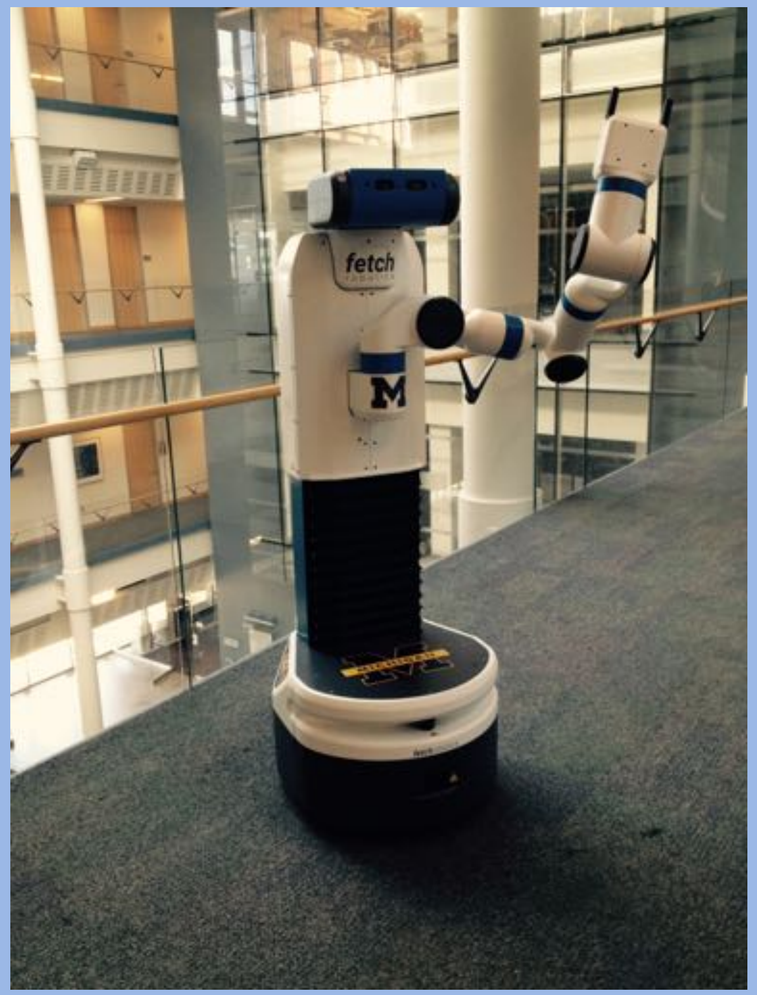
2009

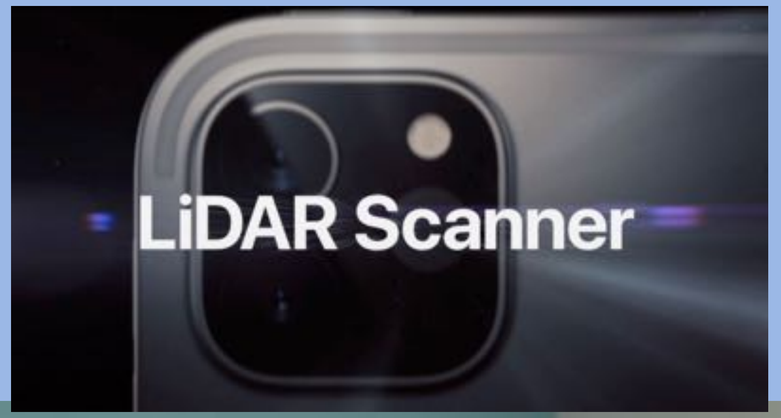
2020

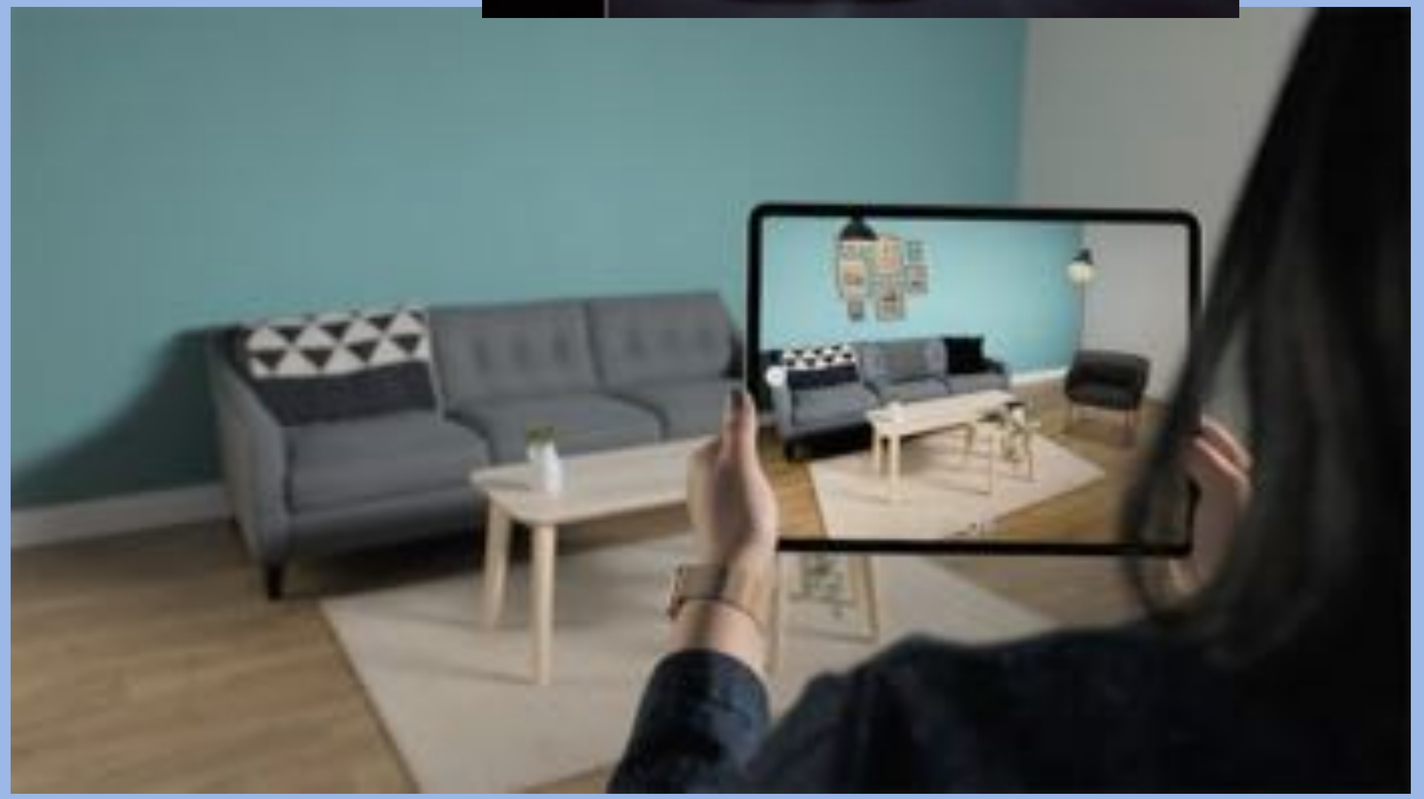




Slide borrowed from Michigan Robotics autorob.org







Slide borrowed from Michigan Robotics autorob.org

Slide borrowed from Michigan Robotics autorob.org

Use any robot x

to perform any task y

in any environment z

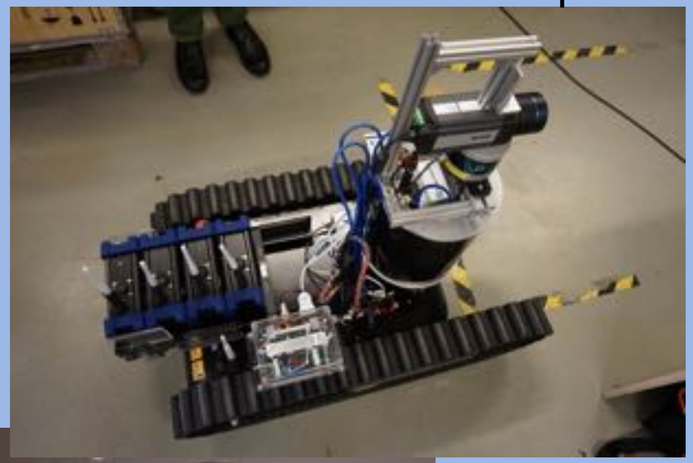
Slide borrowed from Michigan Robotics autorob.org

The 3Ds: Dirty, Dull, and Dangerous

"Autonomous" Driving

Infrastructure inspection

Nuclear cleanup

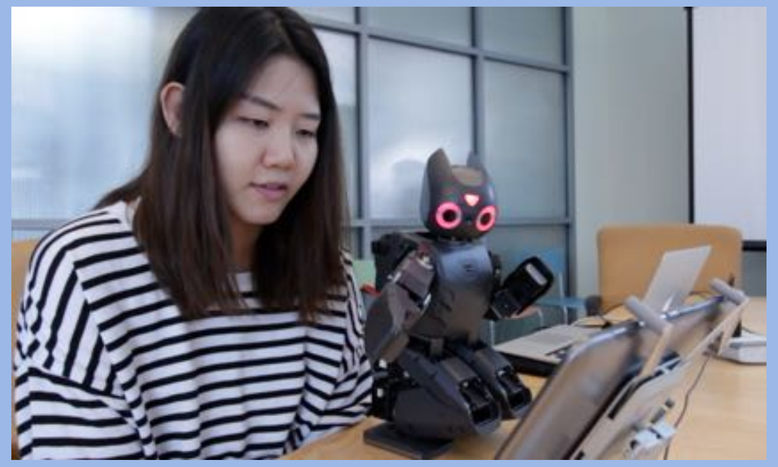


https://www.shadowrobot.com/blog/robots-saving-humans-from-dangerous-jobs/

https://techcrunch.com/2018/06/05/remote-control-driverless-car-startup-partners-with-vehicle-manufacturers/

Social Robotics

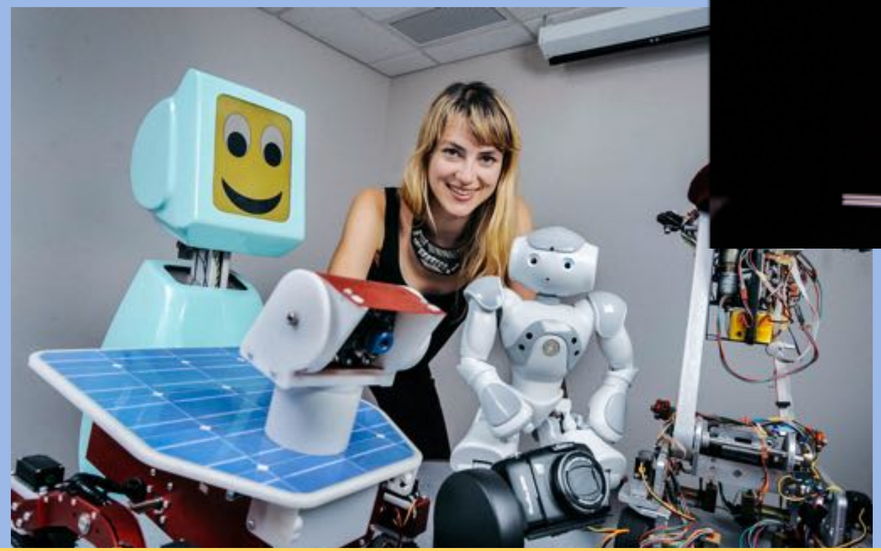
Autism treatment

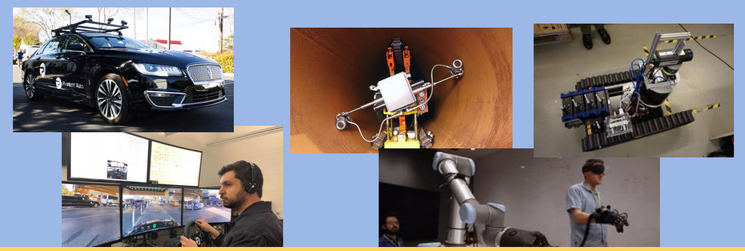


Education

Rehabilitation

Entertainment





Elder care

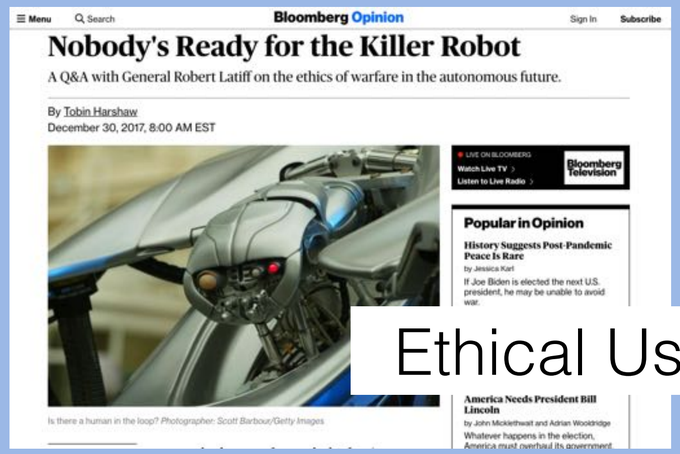
Agriculture

Exploration

Manufacturing

Security

Ethical Use


Menu Search **Bloomberg Opinion** Sign In Subscribe

Nobody's Ready for the Killer Robot

A Q&A with General Robert Latiff on the ethics of warfare in the autonomous future.

By Tobin Harshaw
December 30, 2017, 8:00 AM EST

LIVE ON BLOOMBERG
Watch Live TV
Listen to Live Radio
Bloomberg Television

Popular in Opinion

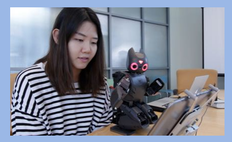
History Suggests Post-Pandemic Peace Is Rare
by Jessica Karl
If Joe Biden is elected the next U.S. president, he may be unable to avoid war.

America Needs President Bill Lincoln
by John Micklethwait and Adrian Wooldridge
Whatever happens in the election,

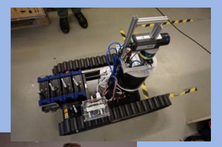
Is there a human in the loop? *Photographer: Scott Barbour/Getty Images*

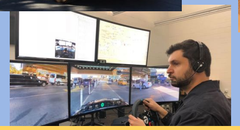
Lethal Force

Social Robotics



Dirty, Dull, Dangerous





Medicine

Slide borrowed from Michigan Robotics autorob.org

Users

Robot Applications

Custom applications,
Taskable autonomy research

Robot Operating System

LCM ROS Player YARP MOOS10

Operating System

Hardware

Slide borrowed from Michigan Robotics autorob.org

Users

Robot Applications

Custom applications,
Taskable autonomy research

Robot Operating System

LCM ROS Player YARP MOOS10

Operating System

Hardware

Slide borrowed from Michigan Robotics autorob.org

Users

Robot Applications

Custom applications,
Taskable autonomy research

Robot Operating System

Build your own Robot OS

Operating System

Hardware

Robot Operating System

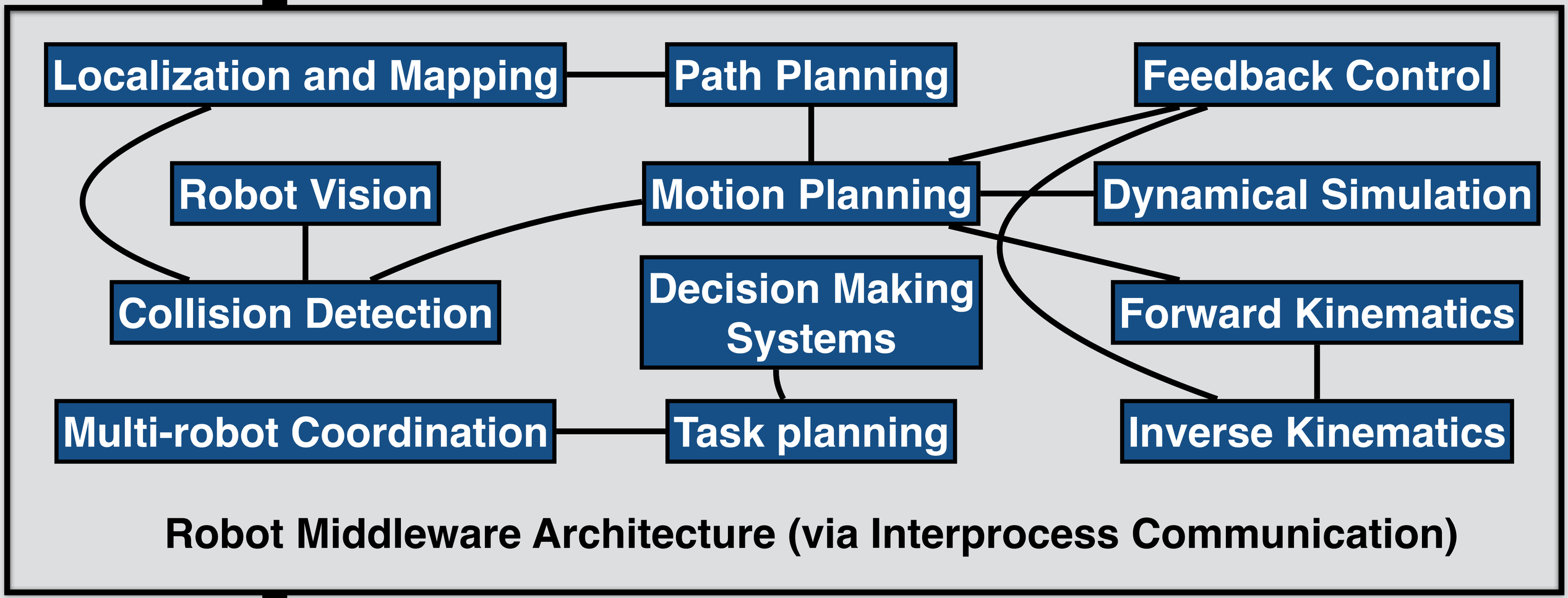
Build your own Robot OS

- Localization and Mapping
- Path Planning
- Feedback Control
- Robot Vision
- Motion Planning
- Dynamical Simulation
- Collision Detection
- Decision Making Systems
- Forward Kinematics
- Multi-robot Coordination
- Task planning
- Inverse Kinematics

Slide borrowed from Michigan Robotics autorob.org

Robot Operating System

Build your own Robot OS

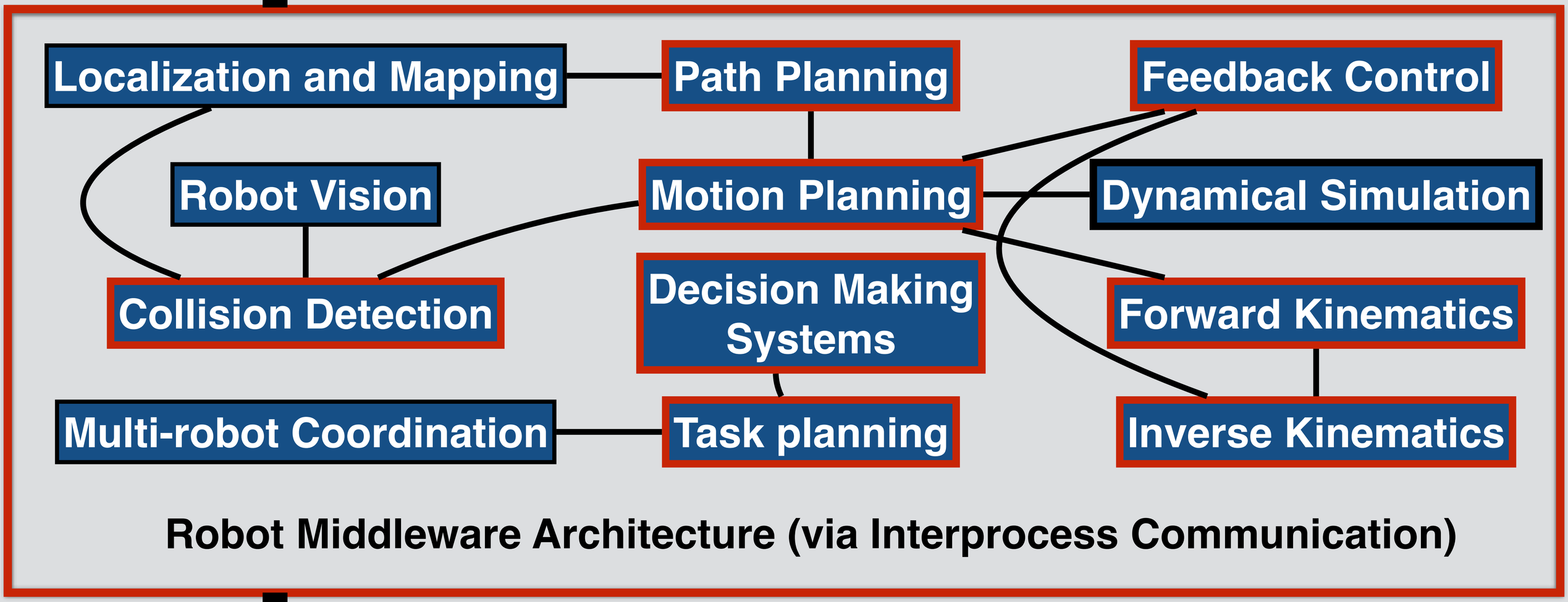


Robot Middleware Architecture (via Interprocess Communication)

Slide borrowed from Michigan Robotics autorob.org

Robot Operating System

Covered at breadth in CSCI5551



Robot Middleware Architecture (via Interprocess Communication)

Slide borrowed from Michigan Robotics autorob.org

Users

Robot Applications

Robot Operating System

Operating System

Hardware

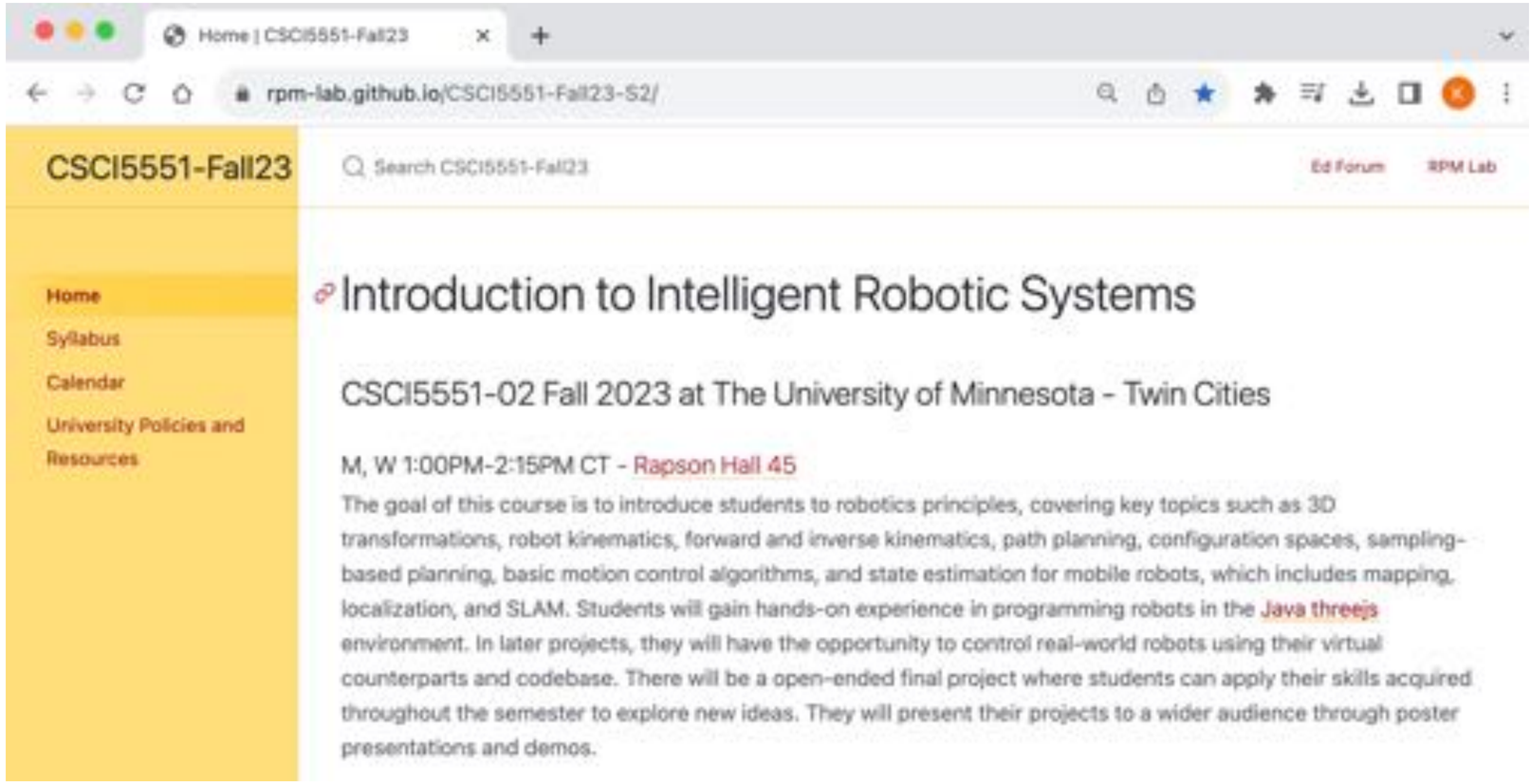
Work with a real robot once this semester

Slide borrowed from Michigan Robotics autorob.org

Course Resources

- Course Website: https://rpm-lab.github.io/CSCI5551-Fall23-S2/
- EdStem: https://edstem.org/us/courses/46318/discussion/
- Canvas: You should see it now!

Course Website



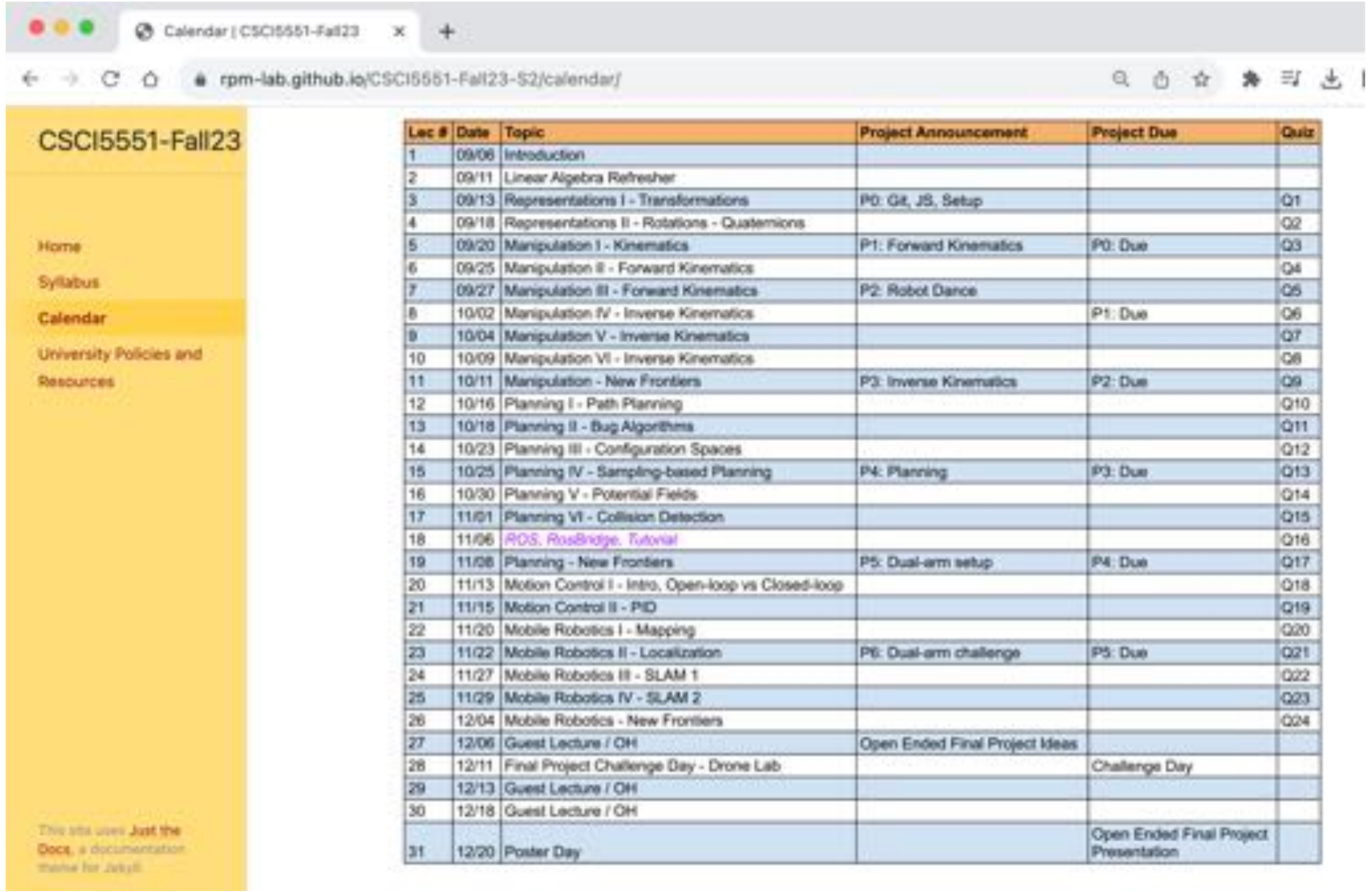
Meeting Logistics

- In-person Lectures
  - Mon & Wed 1:00-2:15 PM Central Time
  - Rapson Hall 45
  - No recordings
- Office Hours
  - Posted on the website, or by appointment

Course Structure

- **Objective:** Give you the computational skills to understand the nuts and bolts of developing a robotic system using kinematics and dynamics.
- **Project focused class:**
  - 7 total projects: building in complexity from basic transformations-rotations to motion planning of a dual-arm robot manipulator

Course Structure



Lec #	Date	Topic	Project Announcement	Project Due	Quiz
1	09/06	Introduction			
2	09/11	Linear Algebra Refresher			
3	09/13	Representations I - Transformations	P0: Git, JS, Setup		Q1
4	09/18	Representations II - Rotations - Quaternions			Q2
5	09/20	Manipulation I - Kinematics	P1: Forward Kinematics	P0: Due	Q3
6	09/25	Manipulation II - Forward Kinematics			Q4
7	09/27	Manipulation III - Forward Kinematics	P2: Robot Dance		Q5
8	10/02	Manipulation IV - Inverse Kinematics		P1: Due	Q6
9	10/04	Manipulation V - Inverse Kinematics			Q7
10	10/09	Manipulation VI - Inverse Kinematics			Q8
11	10/11	Manipulation - New Frontiers	P3: Inverse Kinematics	P2: Due	Q9
12	10/16	Planning I - Path Planning			Q10
13	10/18	Planning II - Bug Algorithms			Q11
14	10/23	Planning III - Configuration Spaces			Q12
15	10/25	Planning IV - Sampling-based Planning	P4: Planning	P3: Due	Q13
16	10/30	Planning V - Potential Fields			Q14
17	11/01	Planning VI - Collision Detection			Q15
18	11/06	ROS, RosBridge, Tutorial			Q16
19	11/08	Planning - New Frontiers	P5: Dual-arm setup	P4: Due	Q17
20	11/13	Motion Control I - Intro, Open-loop vs Closed-loop			Q18
21	11/15	Motion Control II - PID			Q19
22	11/20	Mobile Robotics I - Mapping			Q20
23	11/22	Mobile Robotics II - Localization	P6: Dual-arm challenge	P5: Due	Q21
24	11/27	Mobile Robotics III - SLAM 1			Q22
25	11/29	Mobile Robotics IV - SLAM 2			Q23
26	12/04	Mobile Robotics - New Frontiers			Q24
27	12/06	Guest Lecture / OH	Open Ended Final Project Ideas		
28	12/11	Final Project Challenge Day - Drone Lab		Challenge Day	
29	12/13	Guest Lecture / OH			
30	12/18	Guest Lecture / OH			
31	12/20	Poster Day		Open Ended Final Project Presentation	

Guided Projects

- Project 0
  - Introduction to Git, ThreeJs, Code Stencil
- Project 1
  - Forward Kinematics
- Project 2
  - Robot dance
- Project 3
  - Inverse Kinematics
- Project 4
  - Planning
- Project 5
  - Dual-arm setup
- Project 6
  - Dual-arm Challenge

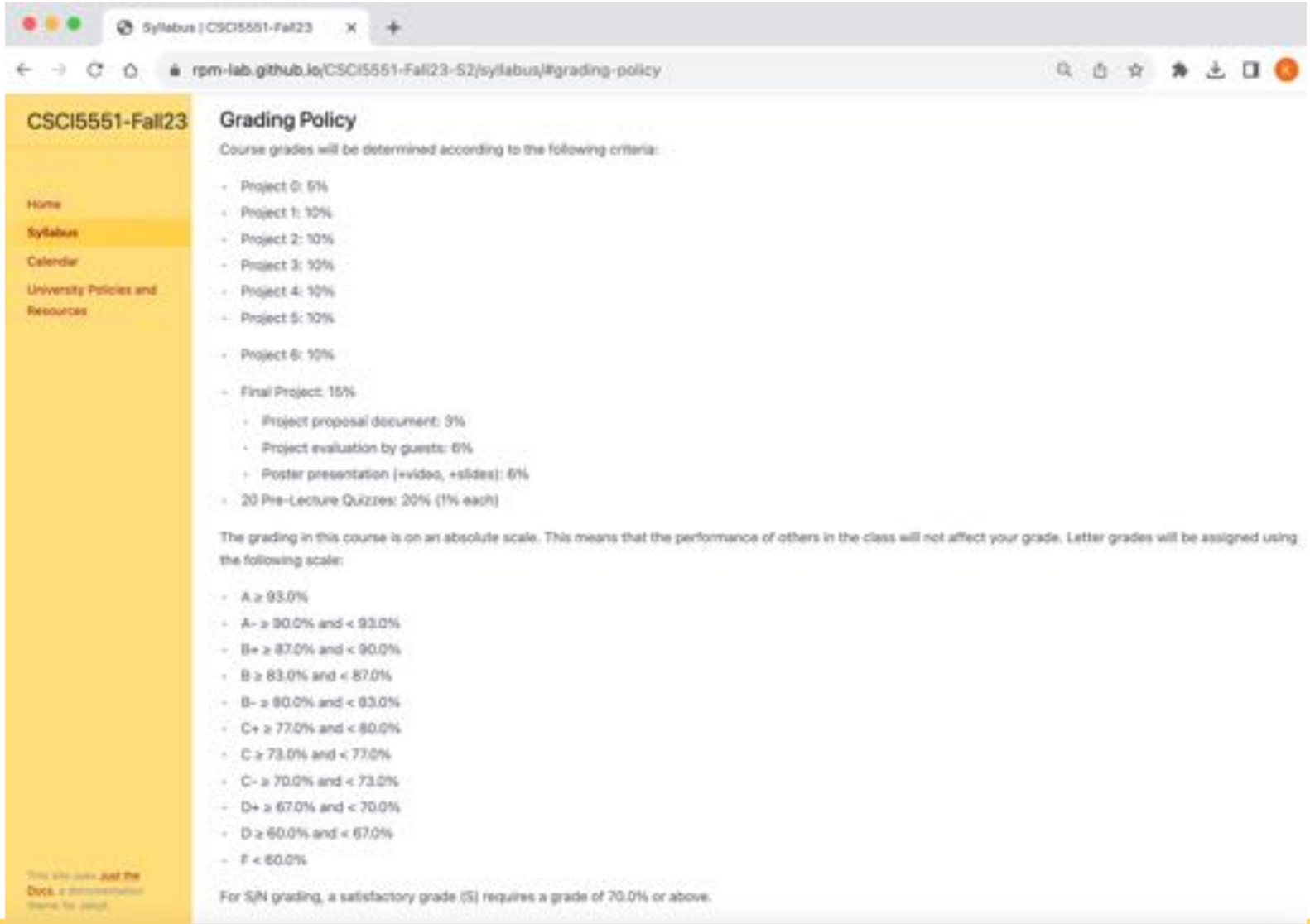
Final Project Challenge

- Open-ended and will let students explore ideas with their learnings from the course.

Project Grading

- Guided Projects 0-6
  - 2 total late days available
  - 25% daily penalty after deadline.
- Open-ended Final Project
  - No late days

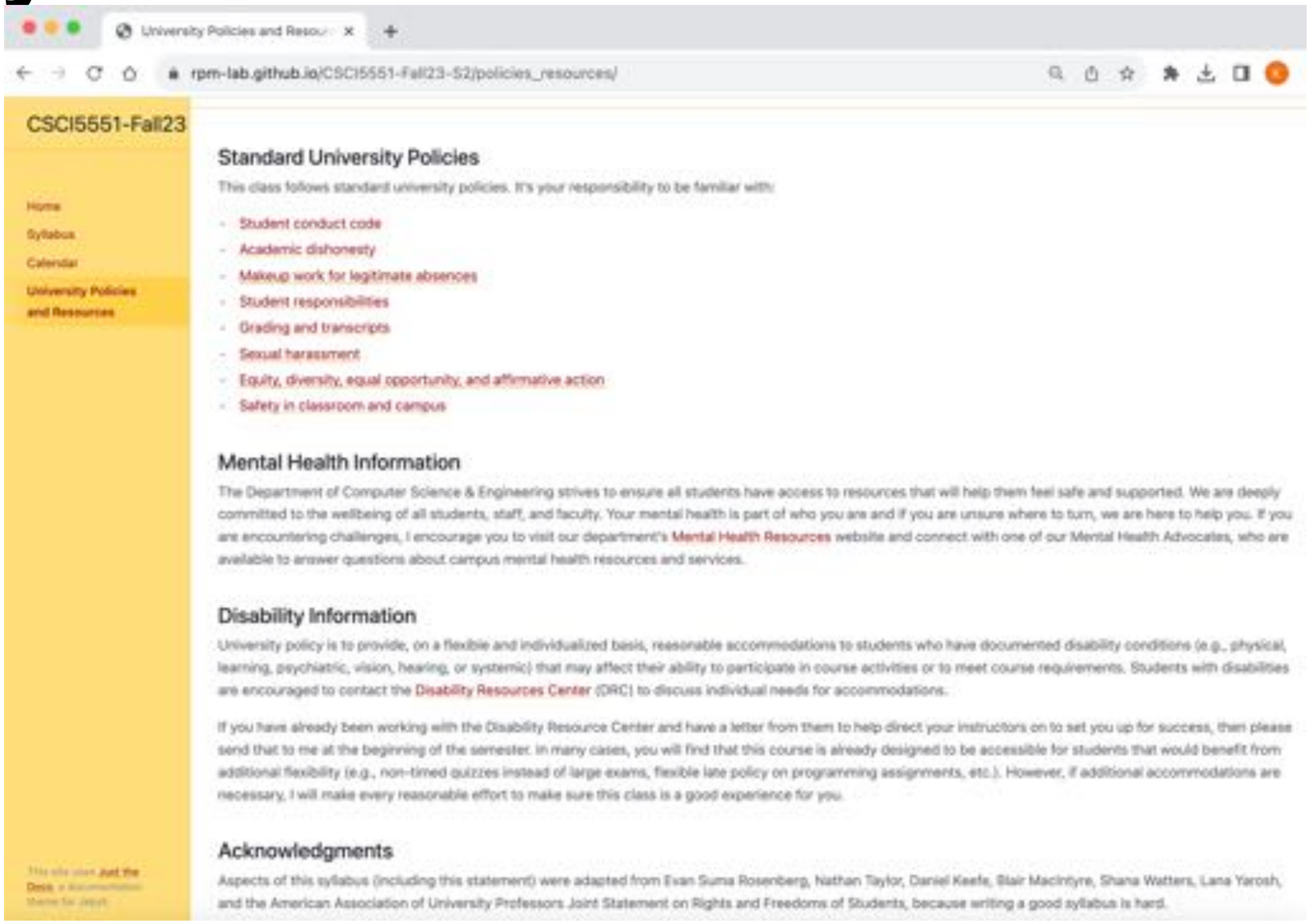
Overall Grading Policy



Collaboration Policy

- All work submitted must be your own.
  - All code submitted must comply with College of Engineering Honor Code.
- Cheating will not be tolerated and can lead to termination from the program.
- No code can be communicated, including verbally.
  - Explicit use of external sources must be clearly cited.
- Free flow of discussion and ideas is encouraged.

University Policy



Discussion Forum

EdStem

- EdStem is the discussion forum used in this course.
- Discussion of quizzes and verbatim code must be private.
- You will be added to it this week.

Next lecture:
Linear Algebra Refresher