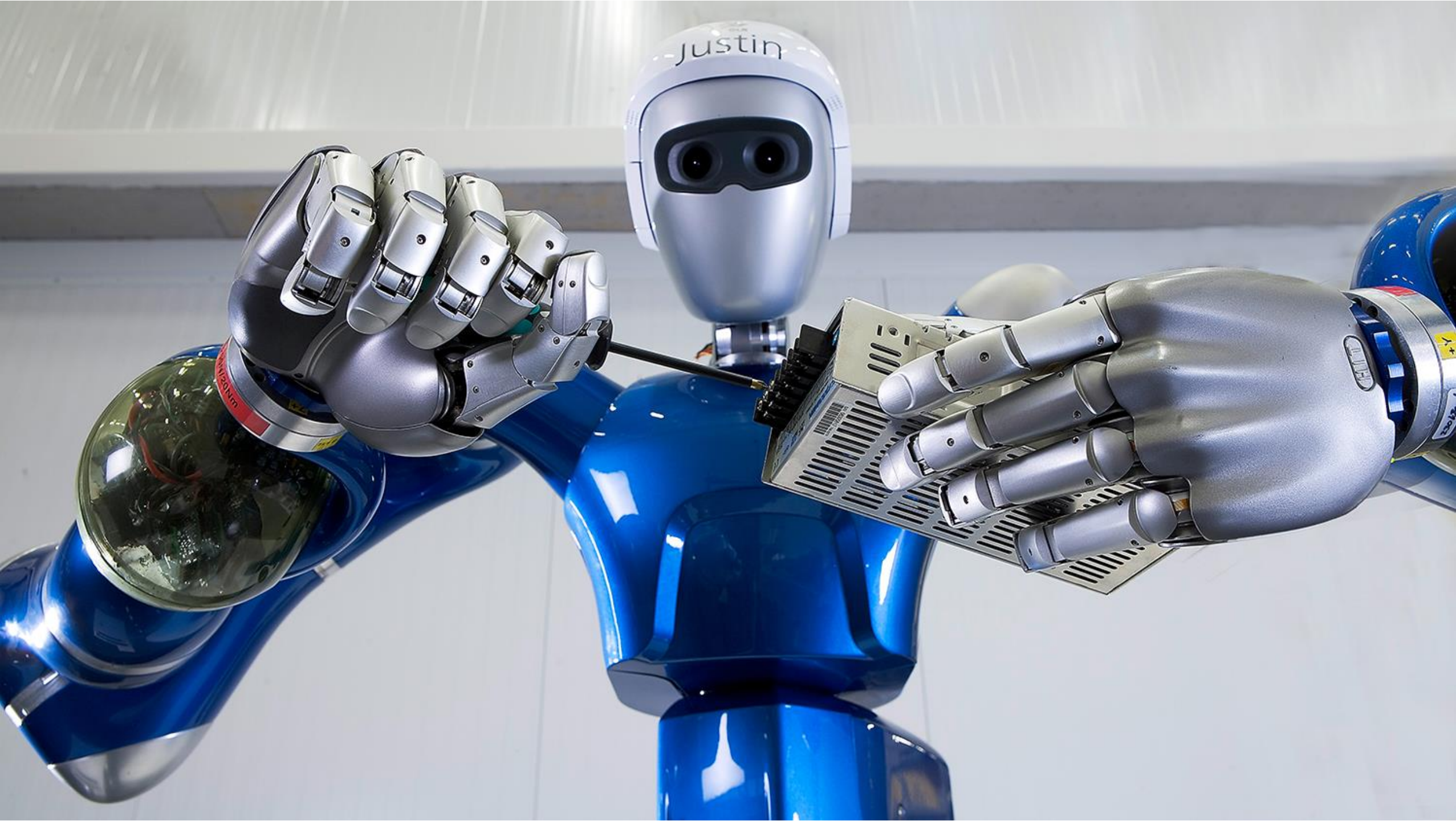
MATLAB EXPO 2019

Developing Autonomous Robots with MATLAB and Simulink

Veer Alakshendra

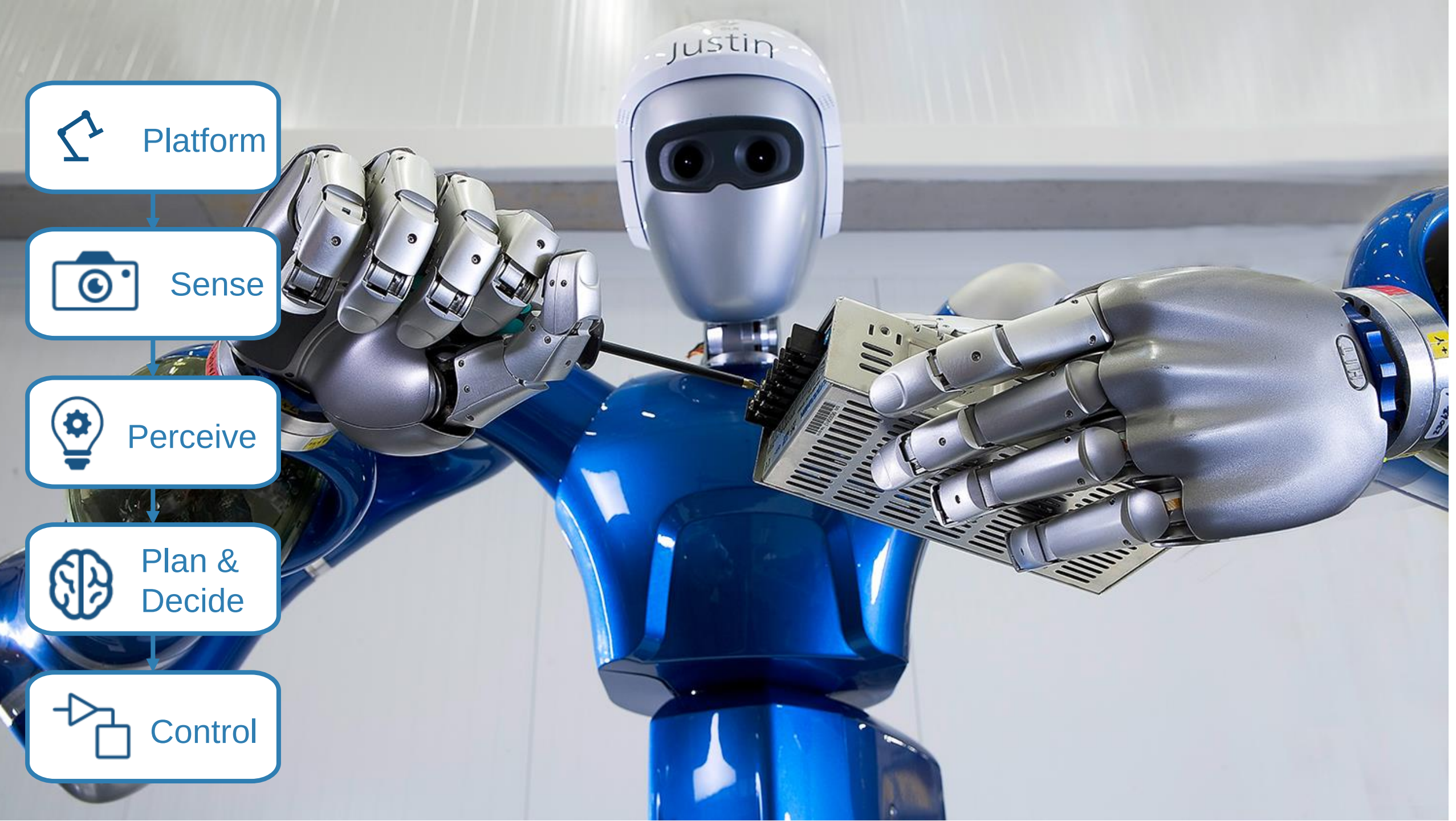

Justin

Challenges with Autonomous Robotics Systems

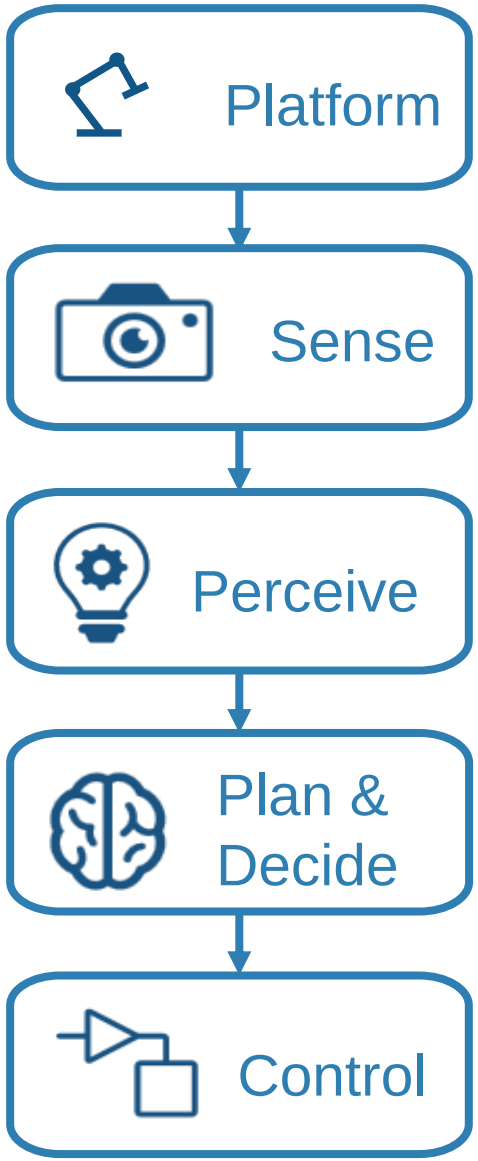
Applying Multidomain Expertise

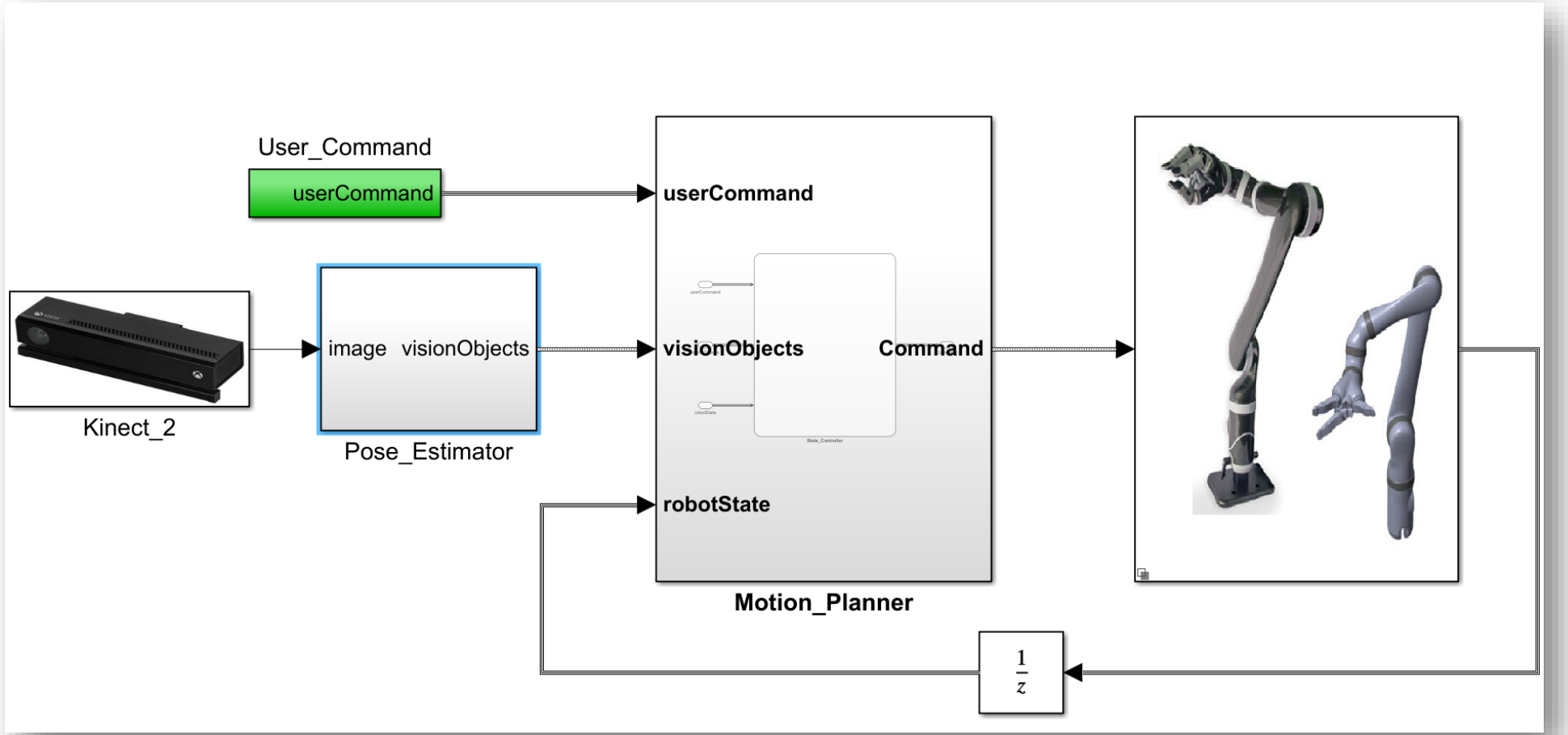
Complexity of Algorithms

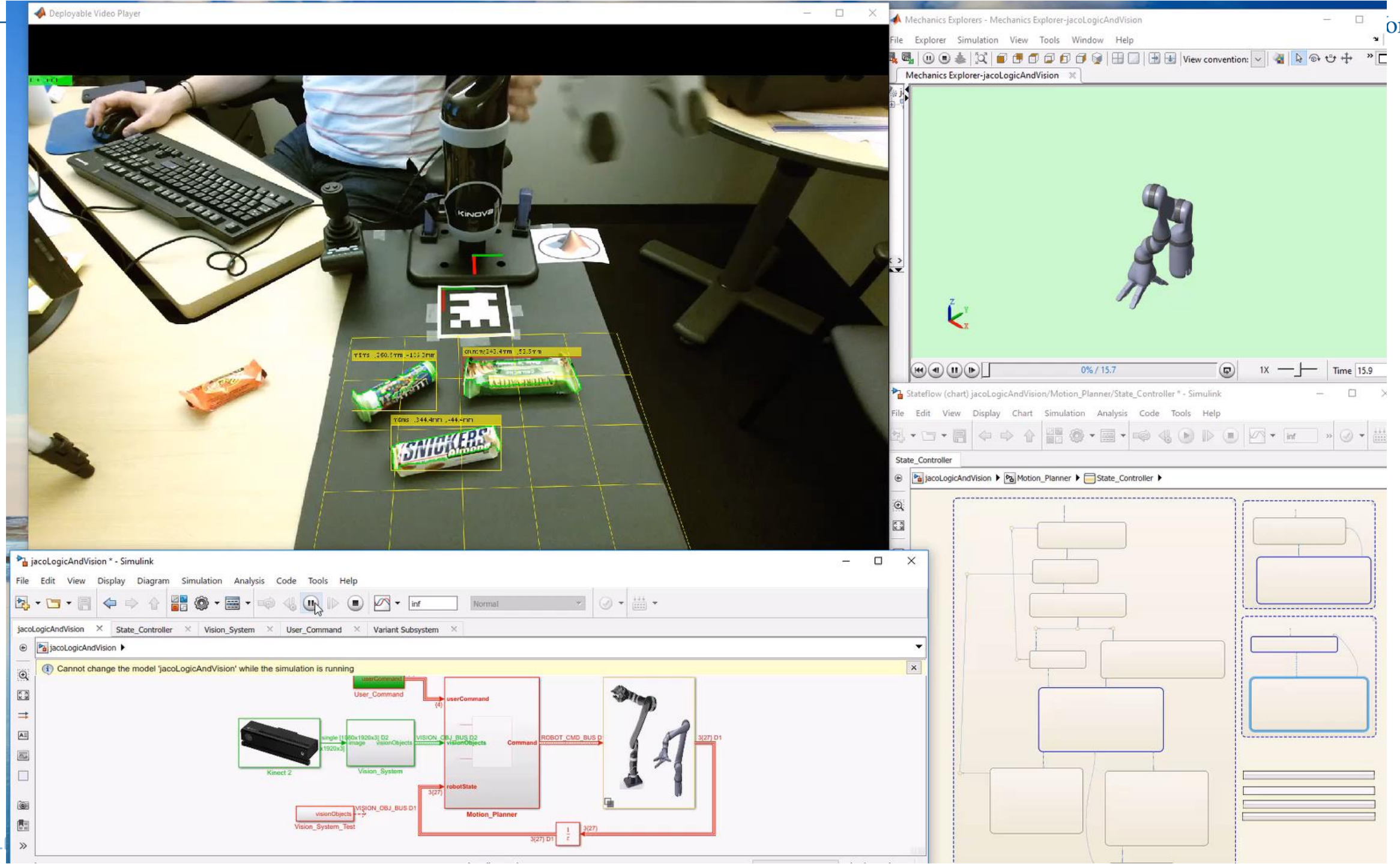
End-to-End workflows


Justin
Platform
Sense
Perceive
Plan &
Decide
Control

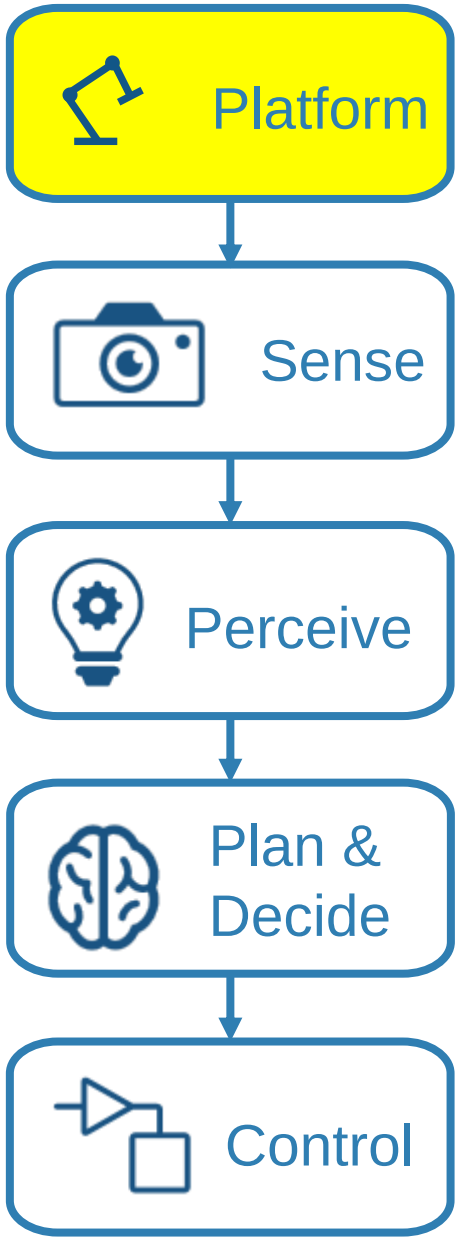
Today: Design Pick and Place Application

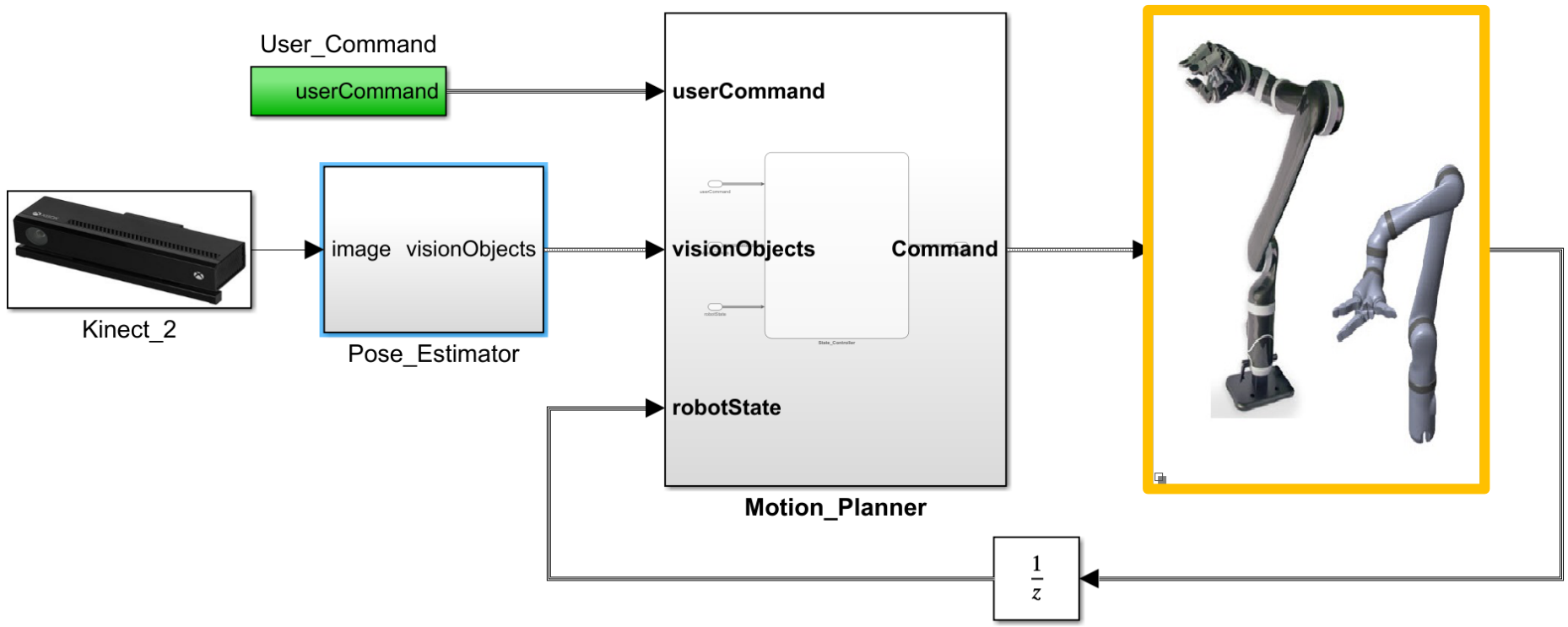





Deployable Video Player
Mechanics Explorers - Mechanics Explorer-jacoLogicAndVision
File Explorer Simulation View Tools Window Help
Mechanics Explorer-jacoLogicAndVision
View convention:
0% / 15.7
1X
Time 15.9
Stateflow (chart) jacoLogicAndVision/Motion_Planner/State_Controller * - Simulink
File Edit View Display Chart Simulation Analysis Code Tools Help
State_Controller
jacoLogicAndVision
Motion_Planner
State_Controller
jacoLogicAndVision * - Simulink
File Edit View Display Diagram Simulation Analysis Code Tools Help
jacoLogicAndVision
State_Controller
Vision_System
User_Command
Variant Subsystem
Cannot change the model 'jacoLogicAndVision' while the simulation is running
User_Command
userCommand
Kinect 2
Vision_System
VISION_OBJ_BUS D2
visionObjects
Command
ROBOT_CMD_BUS D1
3(27) D1
robotState
Motion_Planner
VISION_OBJ_BUS D1
Vision_System_Test
SNICKERS

Today: Design Pick and Place Application





Platform Design

How to create a model of my system that suits my needs?

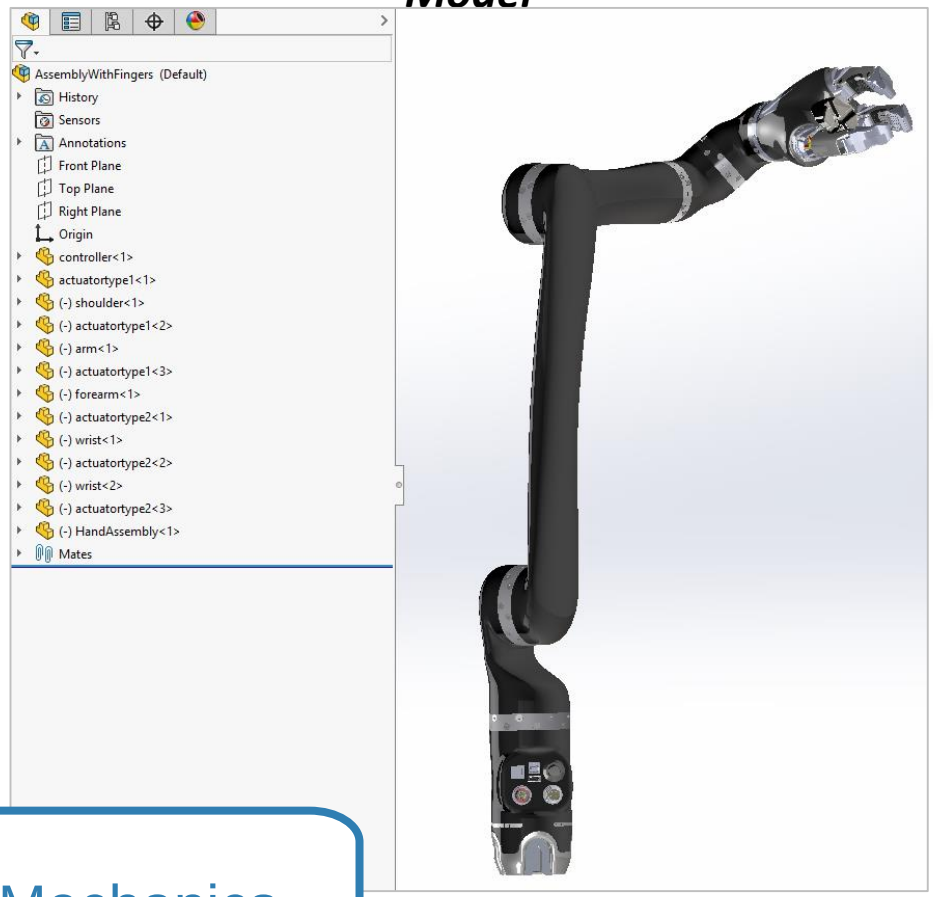
Mechanics

Actuators

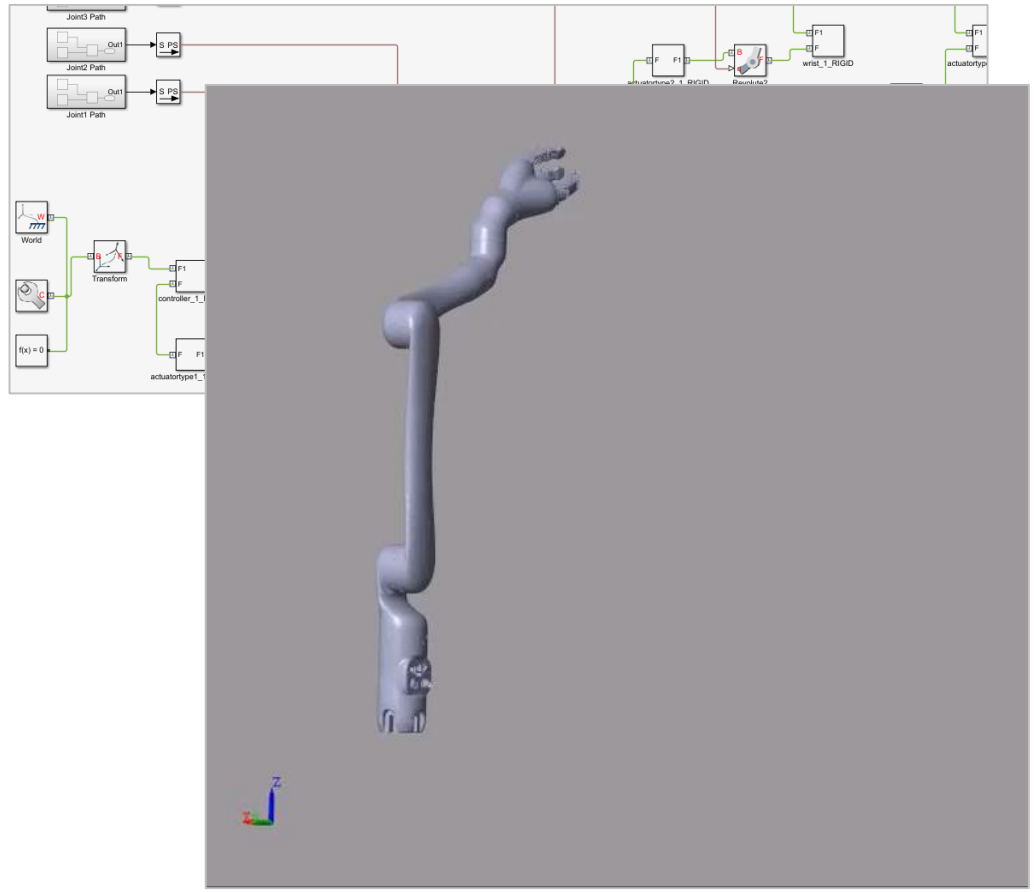
Environment

Import models from common CAD Tools

SolidWorks Model



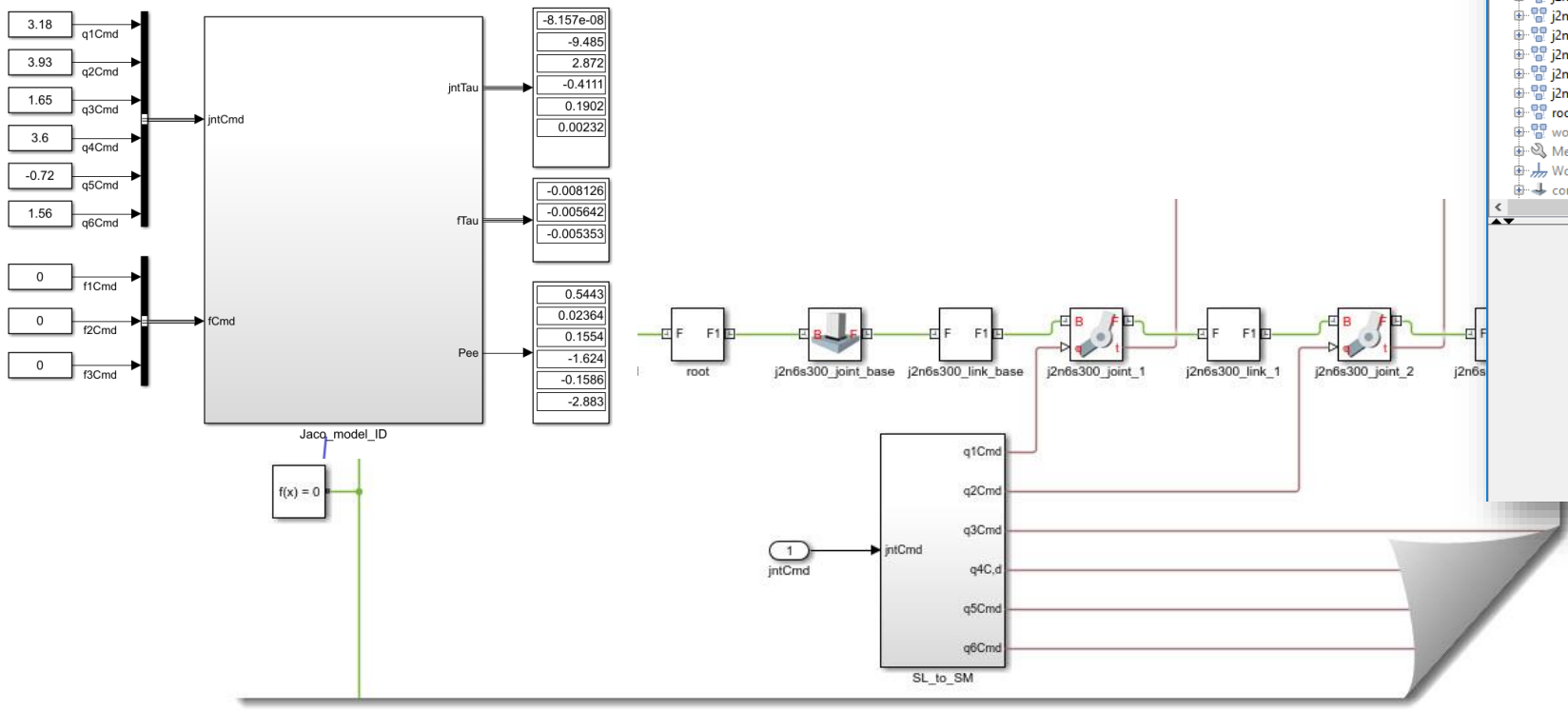
Simscape Multibody Model

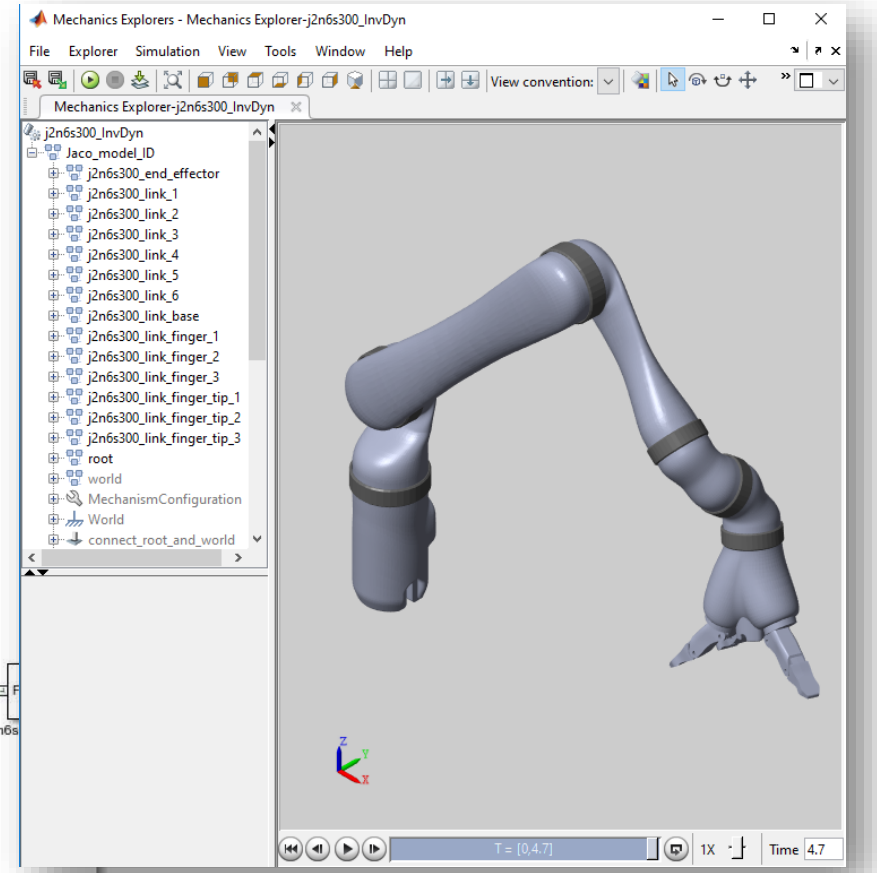


Mechanics

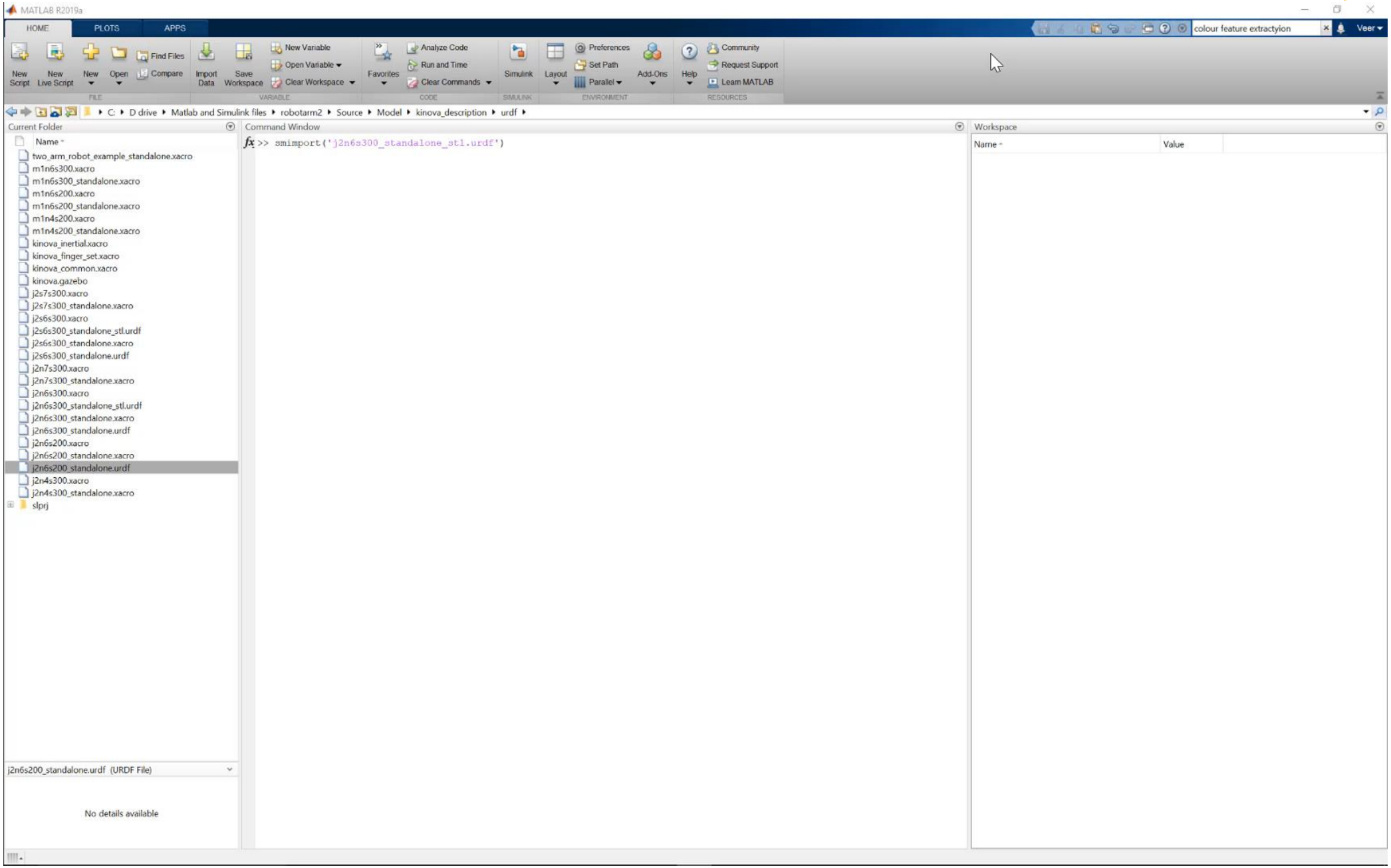
Mechanics: One line import from URDF

```
%% Import robot from URDF
smimport('j2n6s300_standalone_stl.urdf');
```

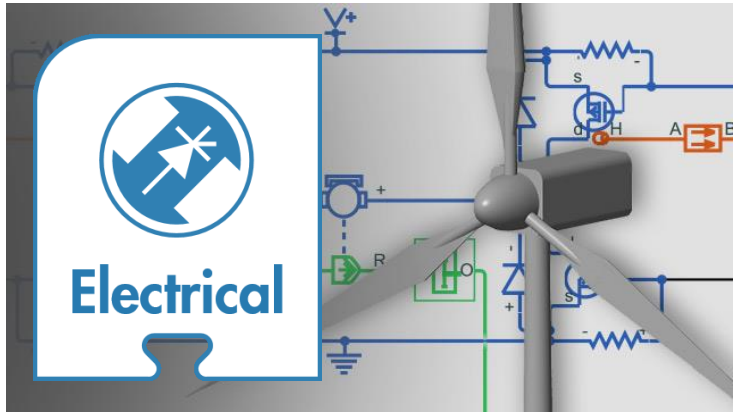


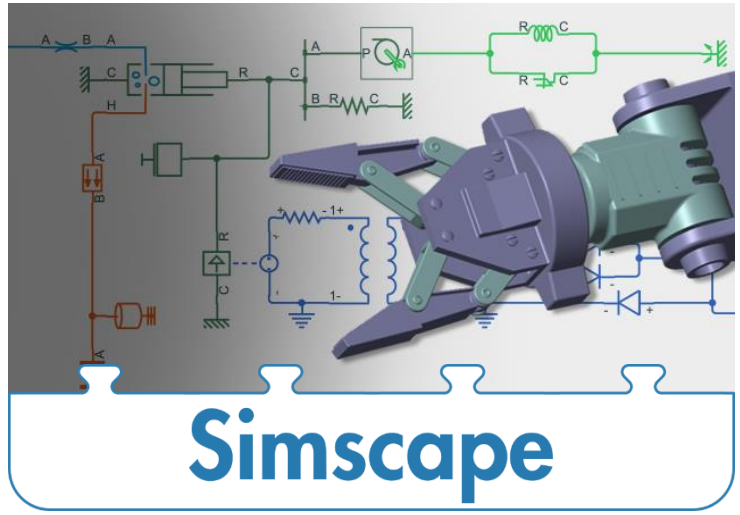


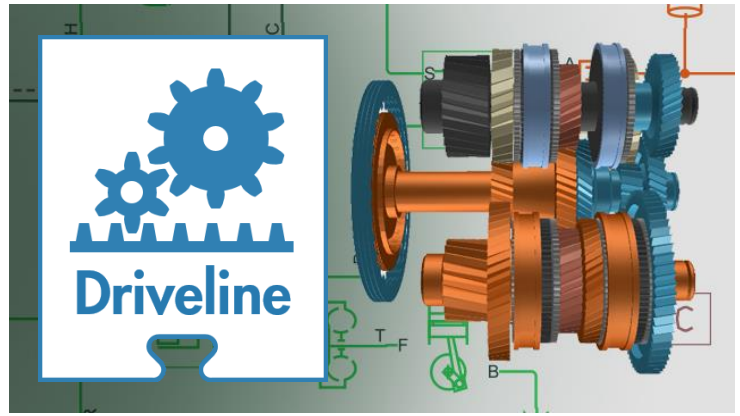
Demo

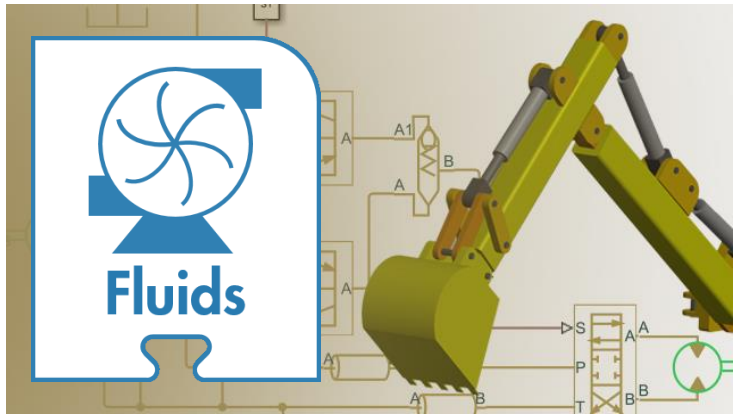

MATLAB R2019a
HOME
PLOTS
APPS
colour feature extractyion
Veer
New Script
New Live Script
New
Open
Find Files
Compare
Import Data
Save Workspace
New Variable
Open Variable
Clear Workspace
Favorites
Analyze Code
Run and Time
Clear Commands
Simulink
Layout
Preferences
Set Path
Parallel
Add-Ons
Help
Community
Request Support
Learn MATLAB
FILE
VARIABLE
CODE
SIMULINK
ENVIRONMENT
RESOURCES
C: ▸ D drive ▸ Matlab and Simulink files ▸ robotarm2 ▸ Source ▸ Model ▸ kinova_description ▸ urdf ▸
Current Folder
Name
two_arm_robot_example_standalone.xacro
m1n6s300.xacro
m1n6s300_standalone.xacro
m1n6s200.xacro
m1n6s200_standalone.xacro
m1n4s200.xacro
m1n4s200_standalone.xacro
kinova_inertial.xacro
kinova_finger_set.xacro
kinova_common.xacro
kinova.gazebo
j2s7s300.xacro
j2s7s300_standalone.xacro
j2s6s300.xacro
j2s6s300_standalone_stl.urdf
j2s6s300_standalone.xacro
j2s6s300_standalone.urdf
j2n7s300.xacro
j2n7s300_standalone.xacro
j2n6s300.xacro
j2n6s300_standalone_stl.urdf
j2n6s300_standalone.xacro
j2n6s300_standalone.urdf
j2n6s200.xacro
j2n6s200_standalone.xacro
j2n6s200_standalone.urdf
j2n4s300.xacro
j2n4s300_standalone.xacro
slprj
j2n6s200_standalone.urdf (URDF File)
No details available
Command Window
>> smimport('j2n6s300_standalone_stl.urdf')
Workspace
Name
Value

Actuators: Model other domains

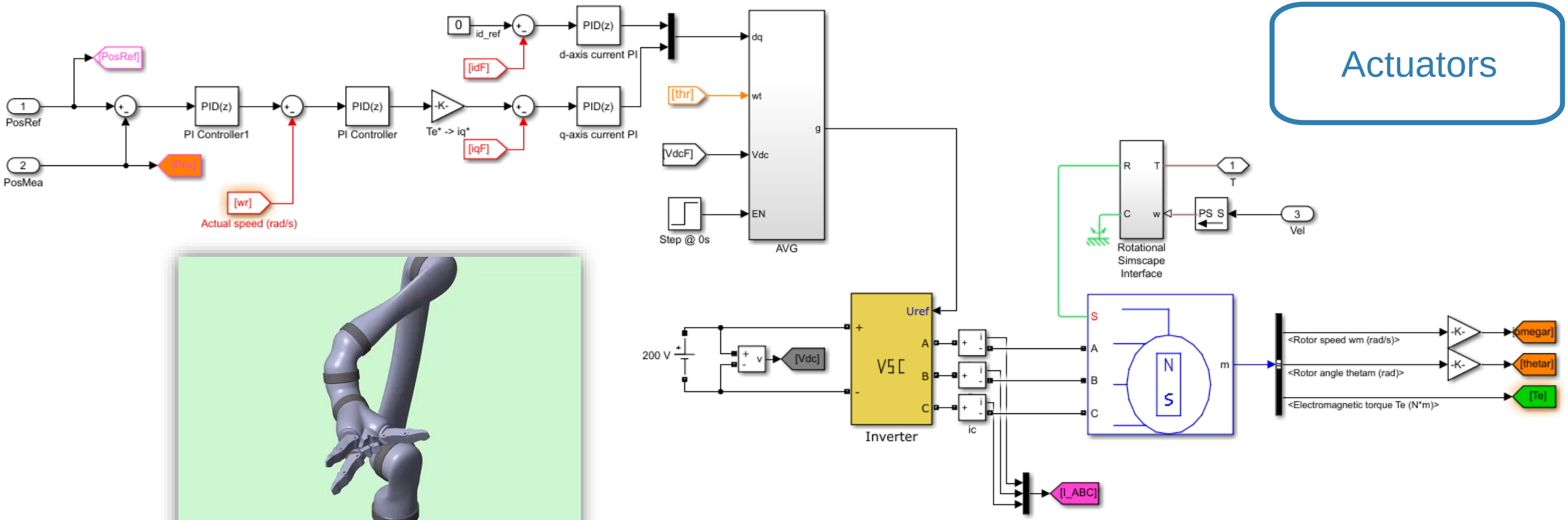




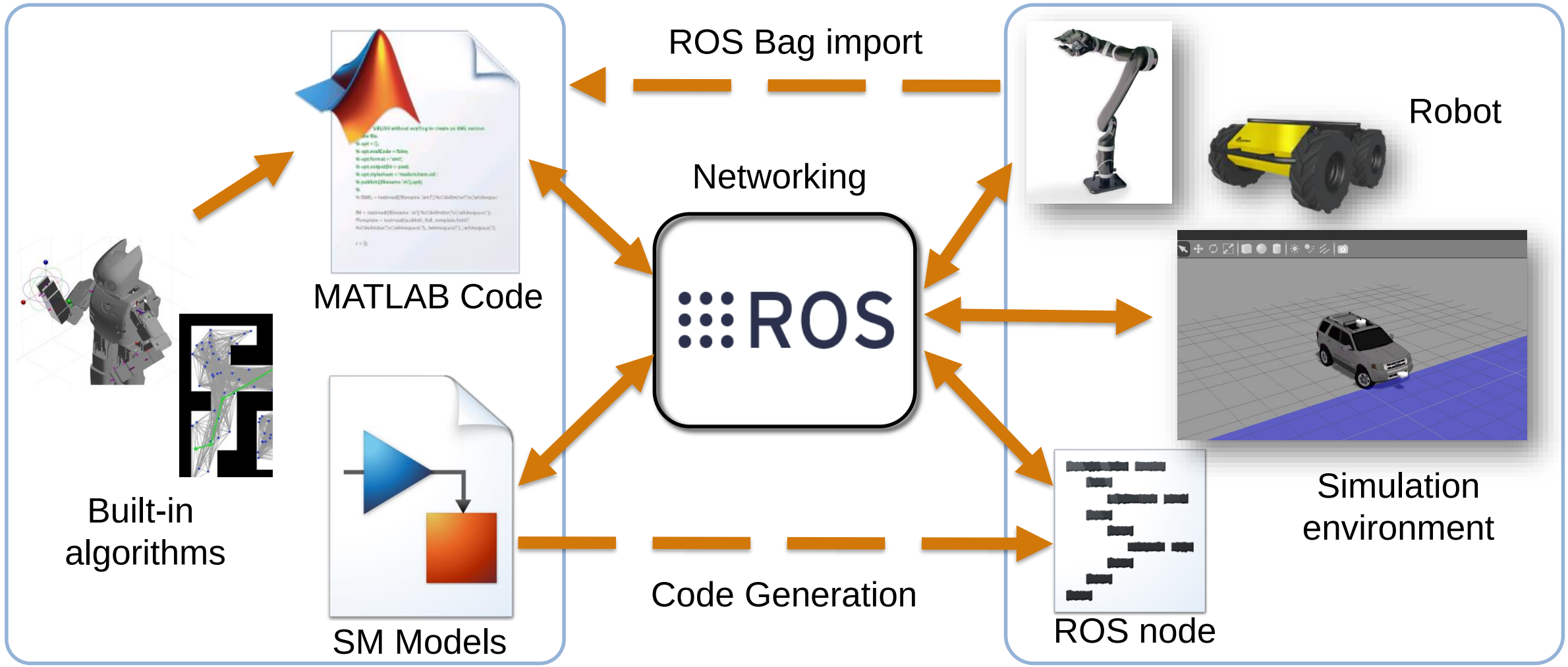




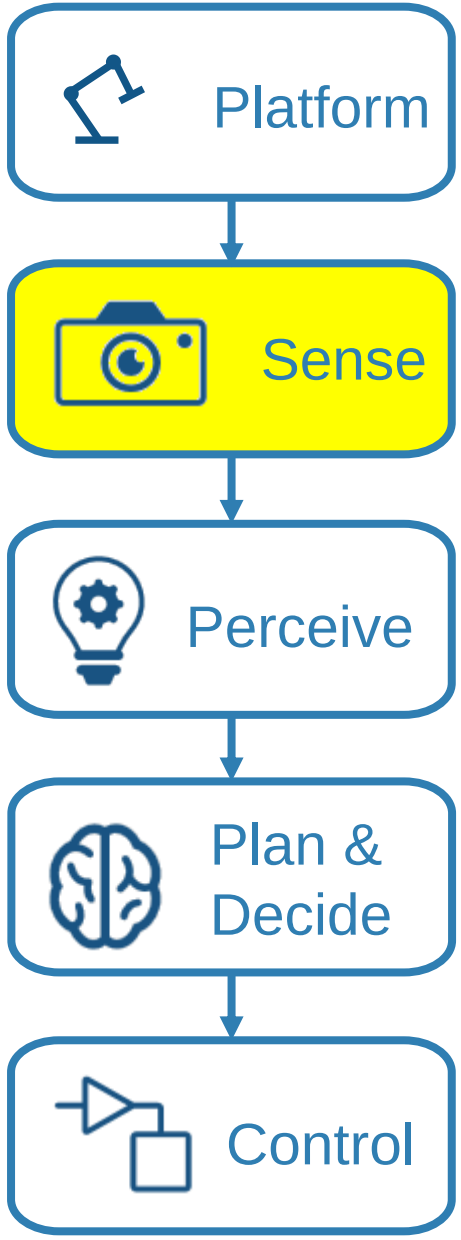
Modeling Actuators

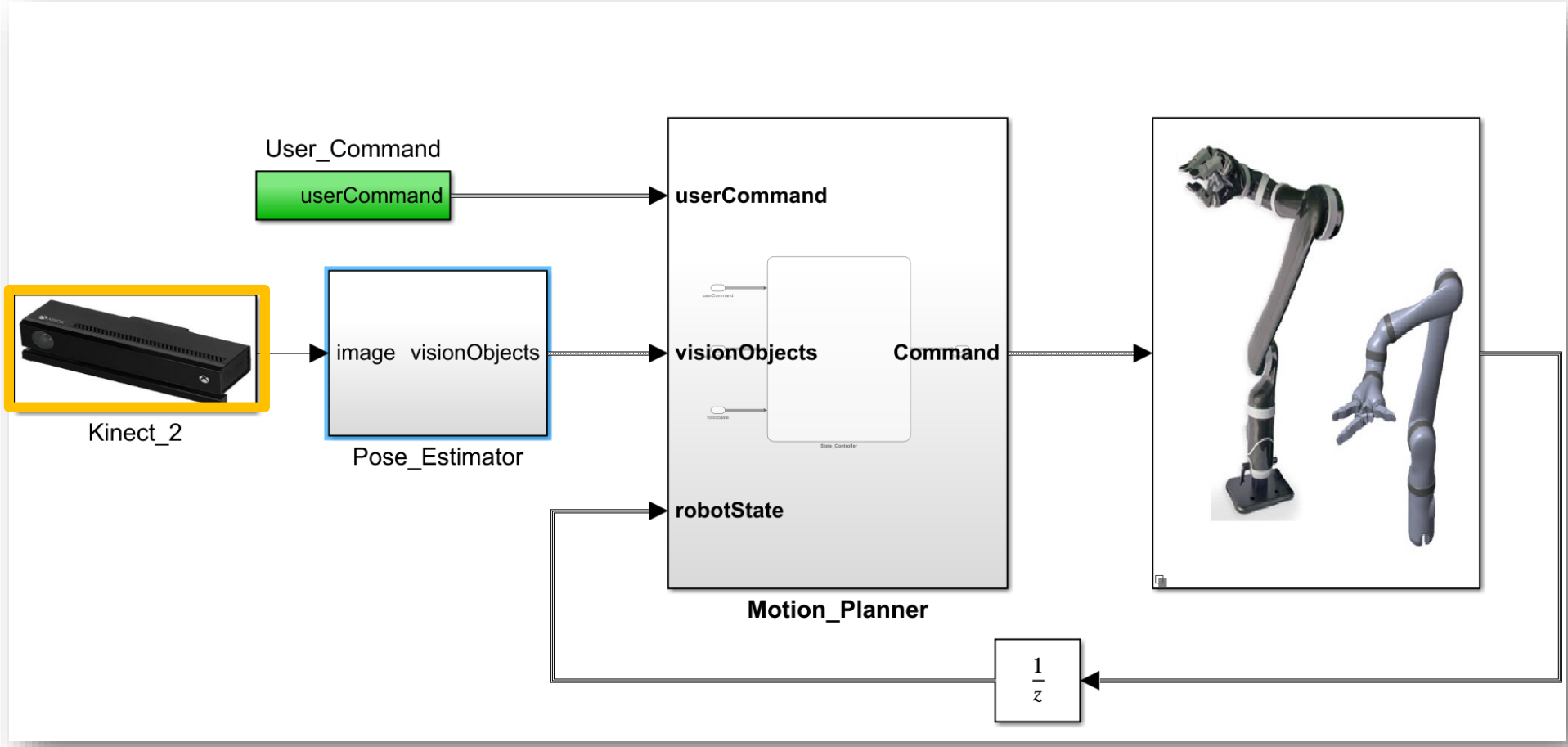


Environment: Connect MATLAB and Simulink with ROS



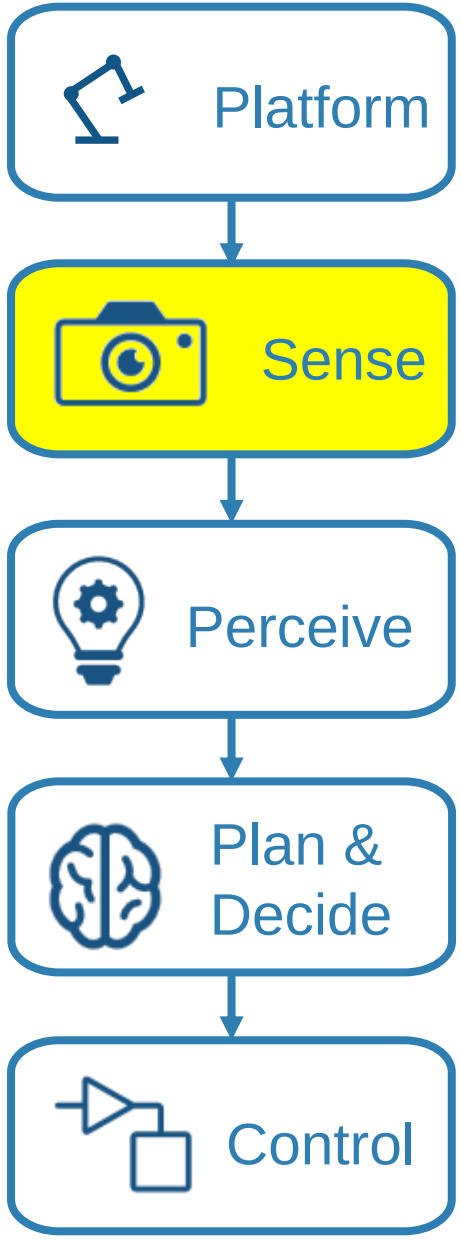
Design Pick and Place Application



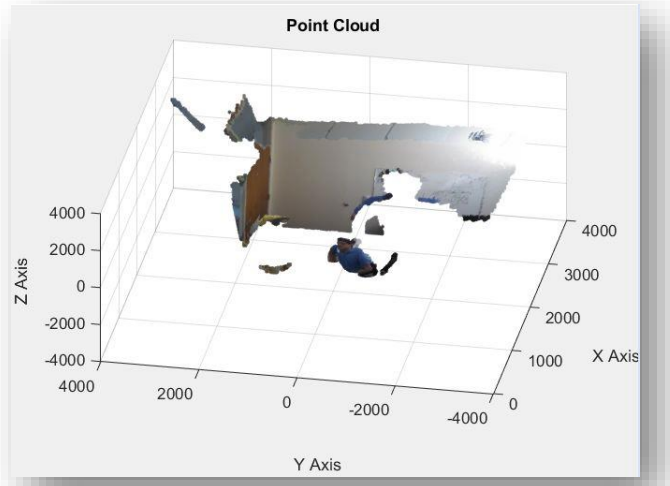


Demo

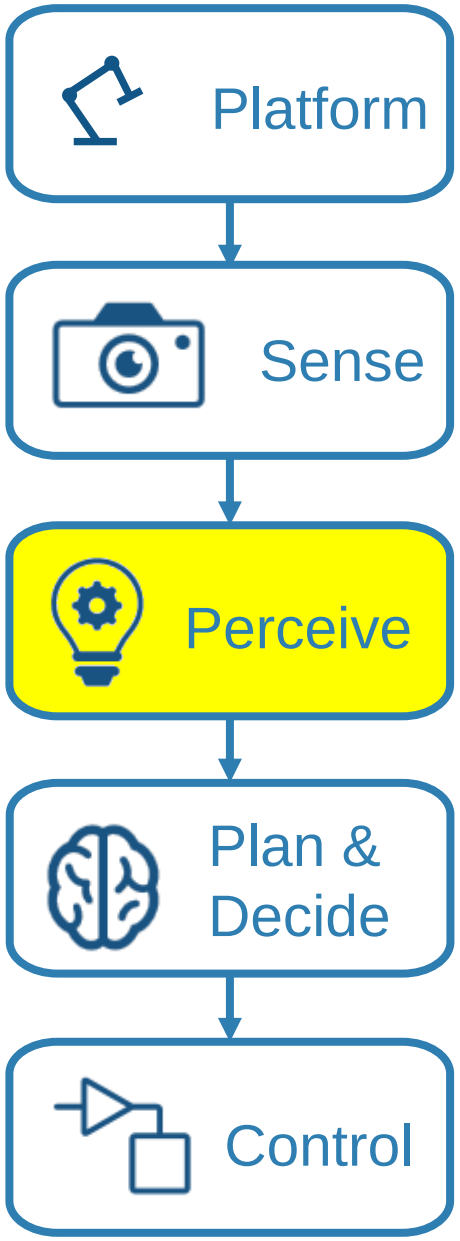
Design Pick and Place Application

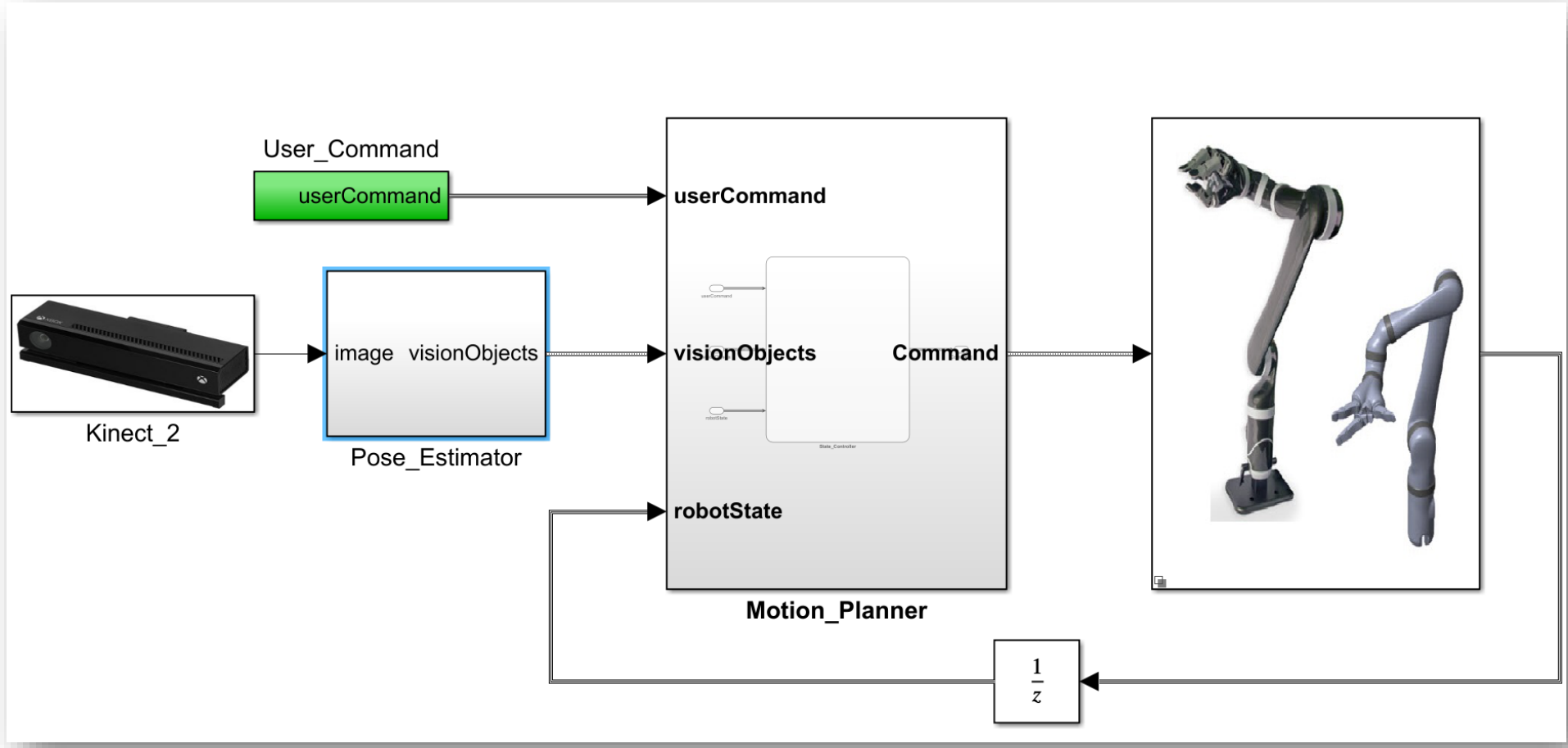


- **Support for Common Sensors**
- **Image analysis**
- **Apps**
- **Image enhancement**
- **Visualizing Point Clouds**



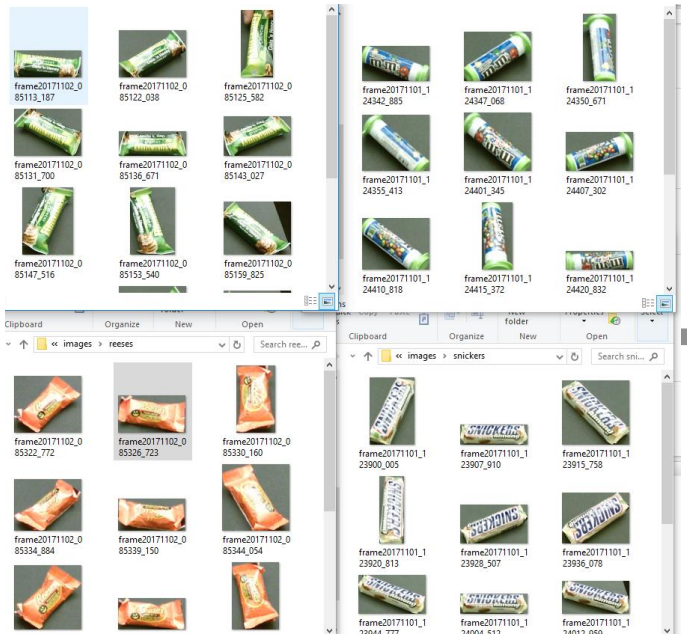
Today: Design Pick and Place Application





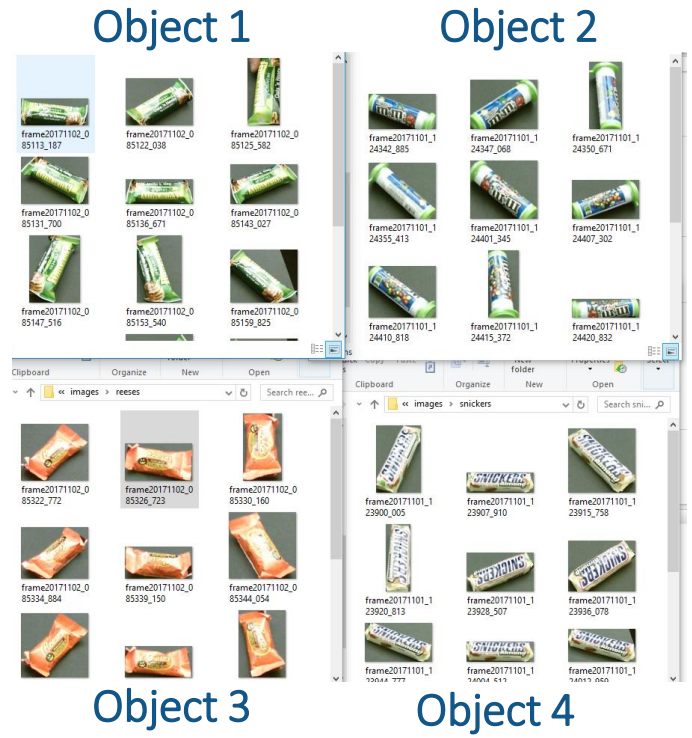
Object Classifier and Pose Estimator

Images



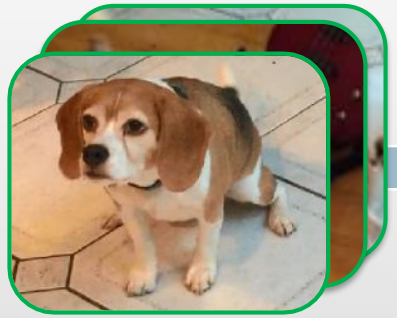
Pose Estimator

Labels and Poses

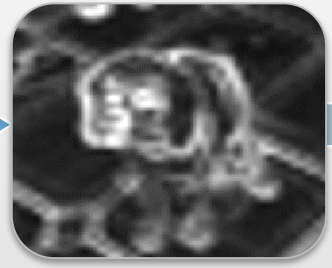


MATLAB makes machine learning easy and accessible

Traditional Machine Learning approach



Traditional Feature Extraction



Classification

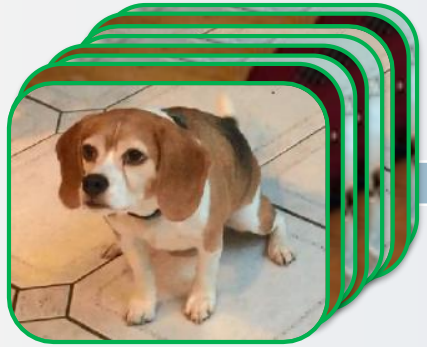
Machine Learning

Dog ✓

Boy ✗

Bicycle ✗

Deep Learning approach



Convolutional Neural Network (CNN)

Learned features

End-to-end learning

Feature learning + Classification

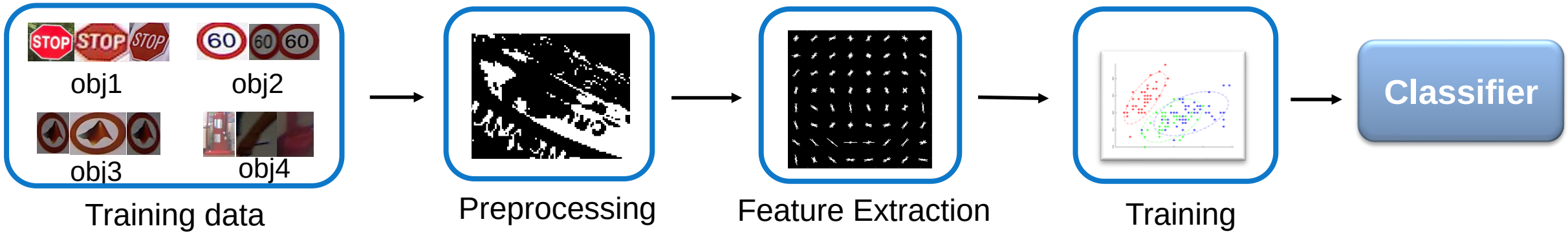
Dog ✓

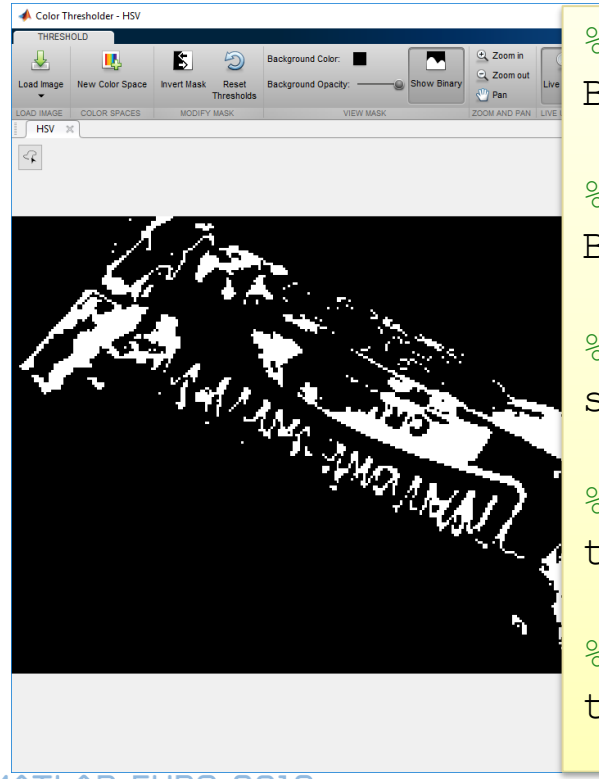
Boy ✗

Bicycle ✗

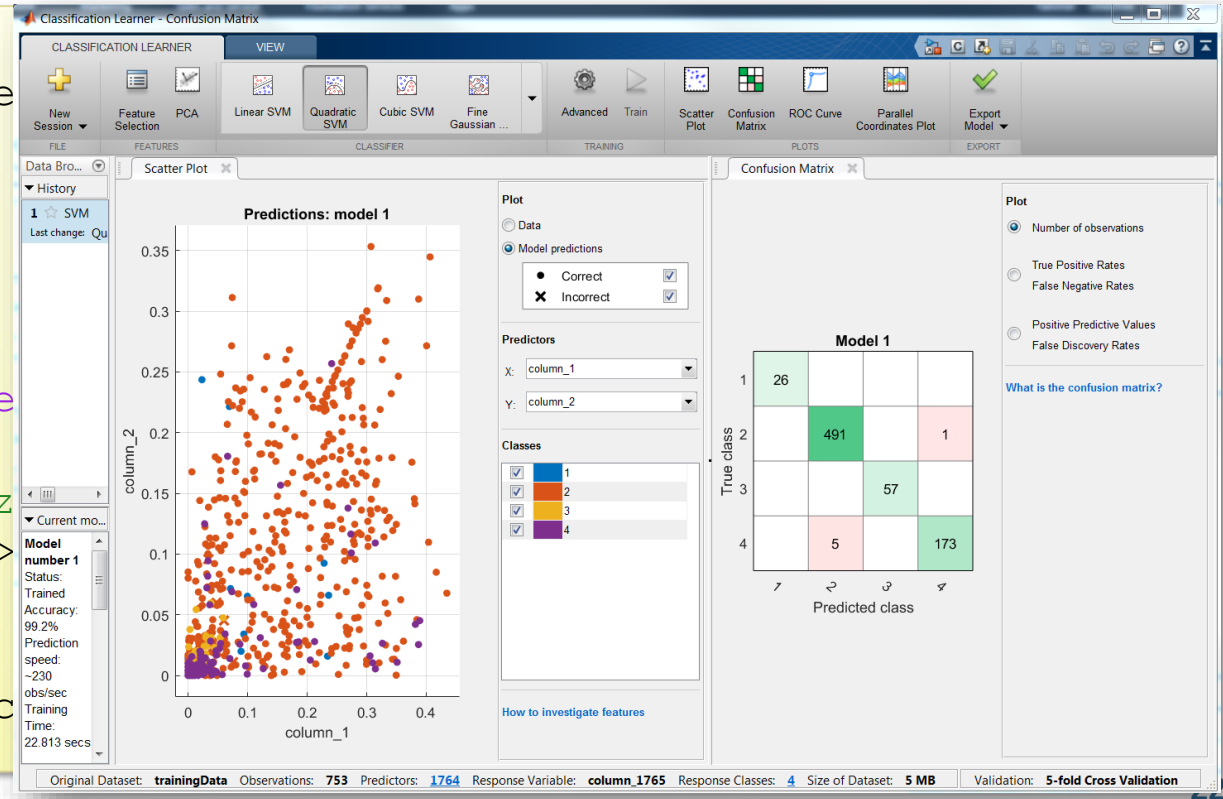
Demo

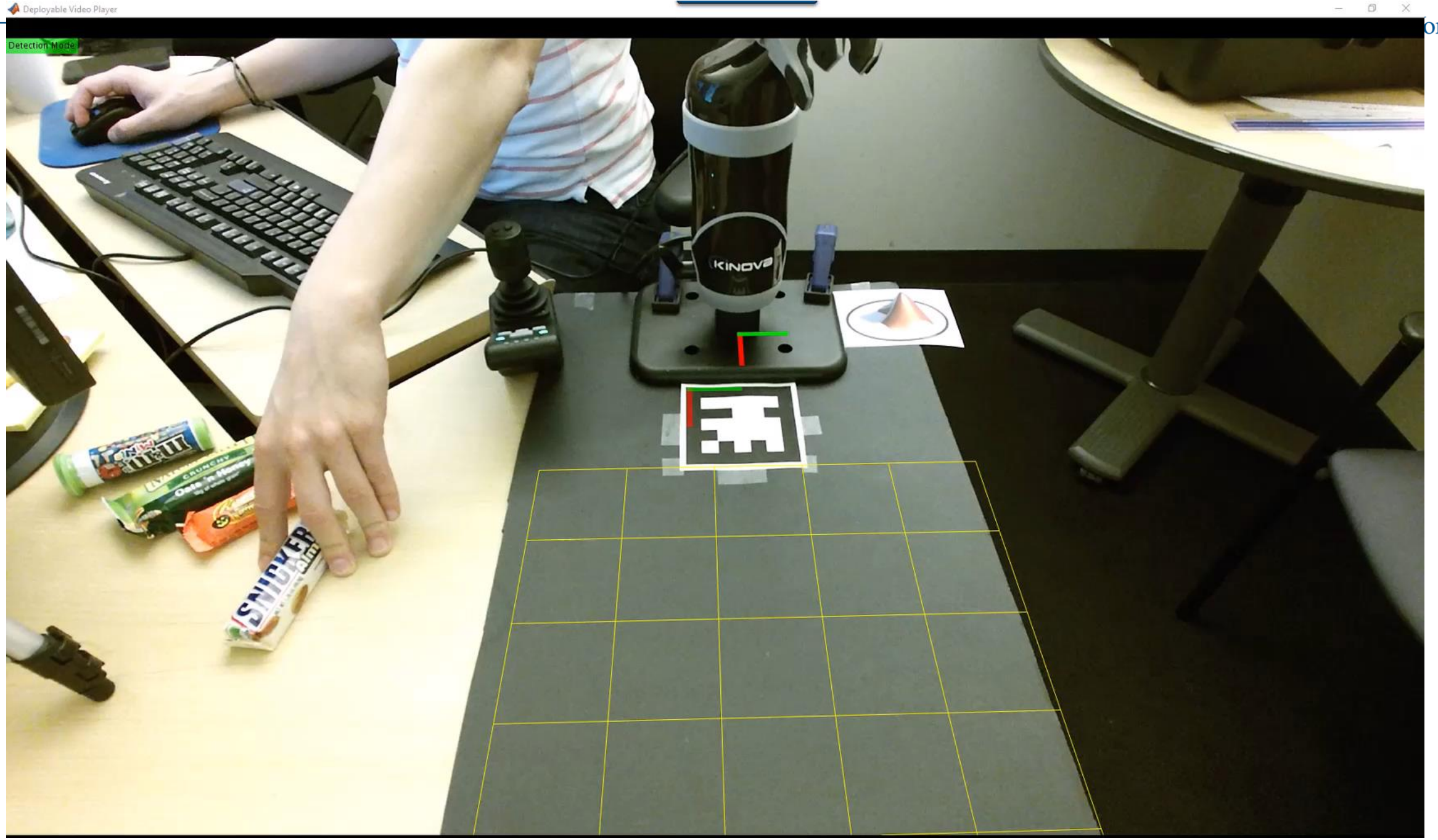
Complex workflows made easy with MATLAB



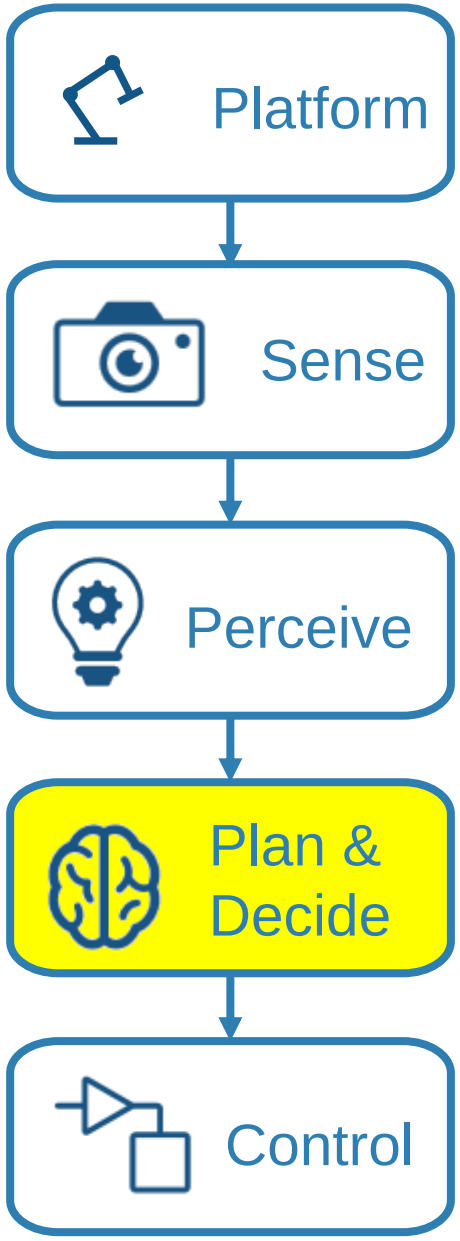


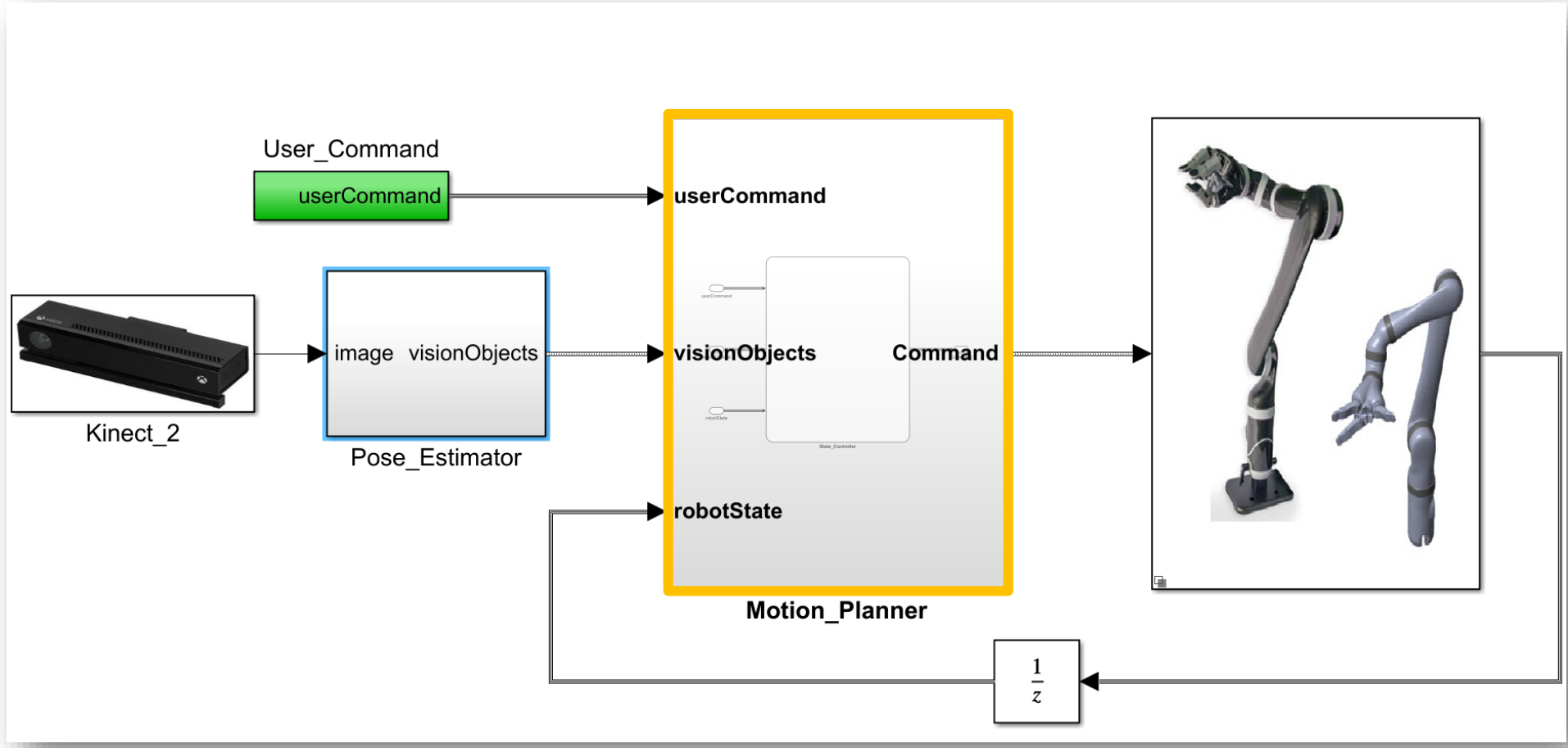

```
% Detect regions
BW = createMask(videoFrame

% Fill image regions
BW = imfill(BW,'holes');

% Get bounding boxes
stats = regionprops('table

% Filter based on area siz
targetIndex = stats.Area >

% Get bounding boxes from
testFeatures(k,:) = extrac
```



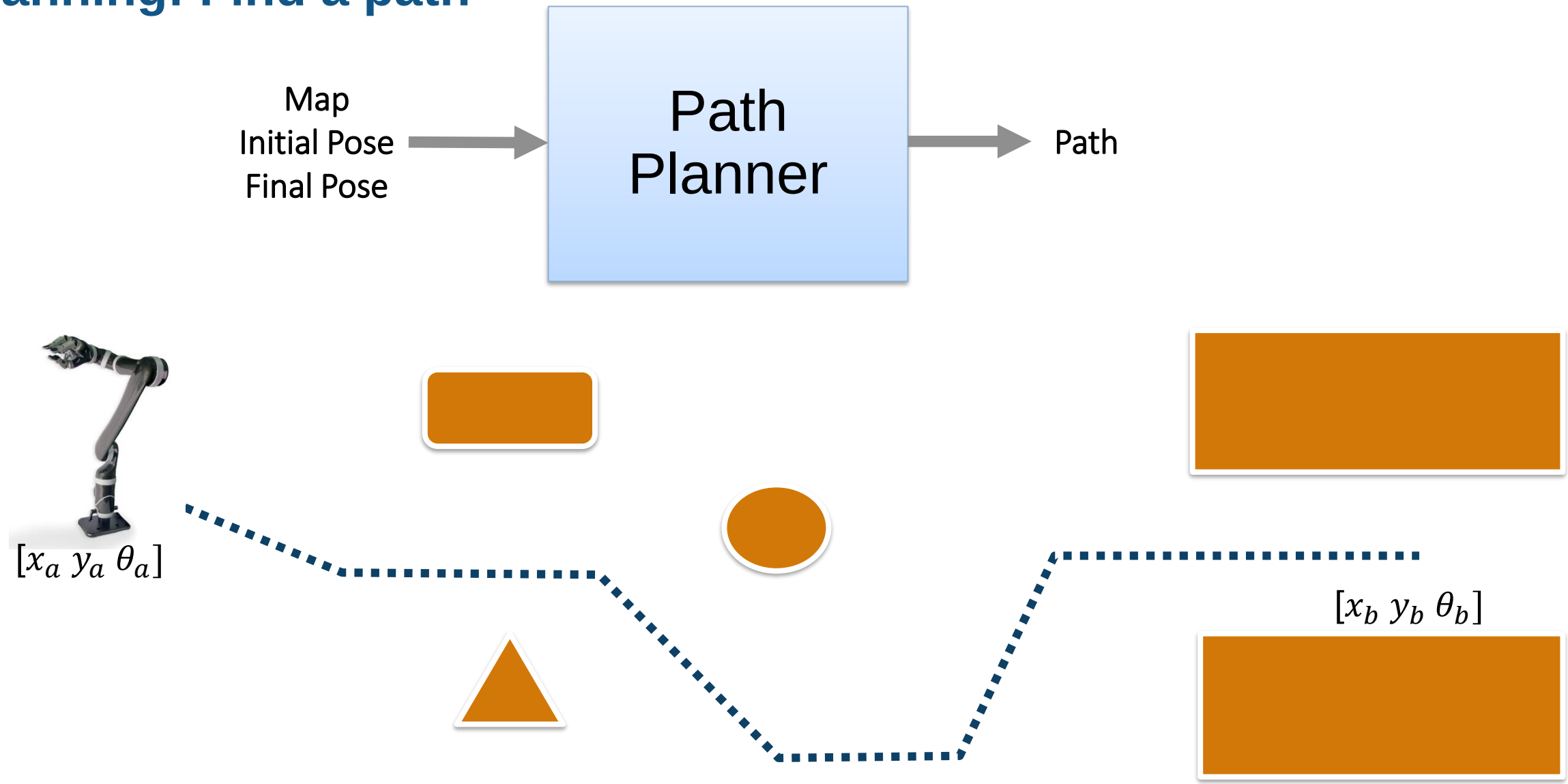

Deployable Video Player
Detection Mode
KINOVA

Design Pick and Place Application

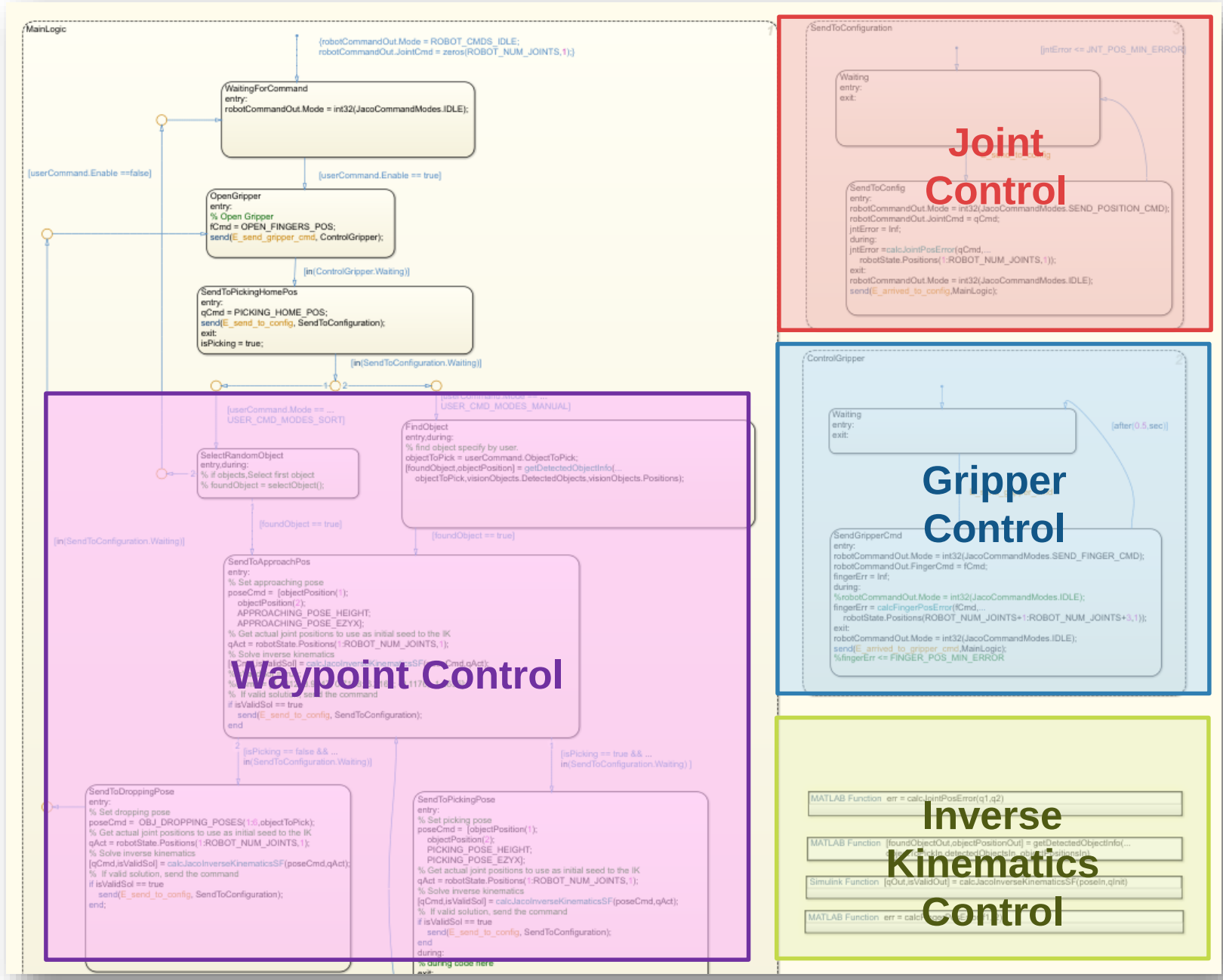




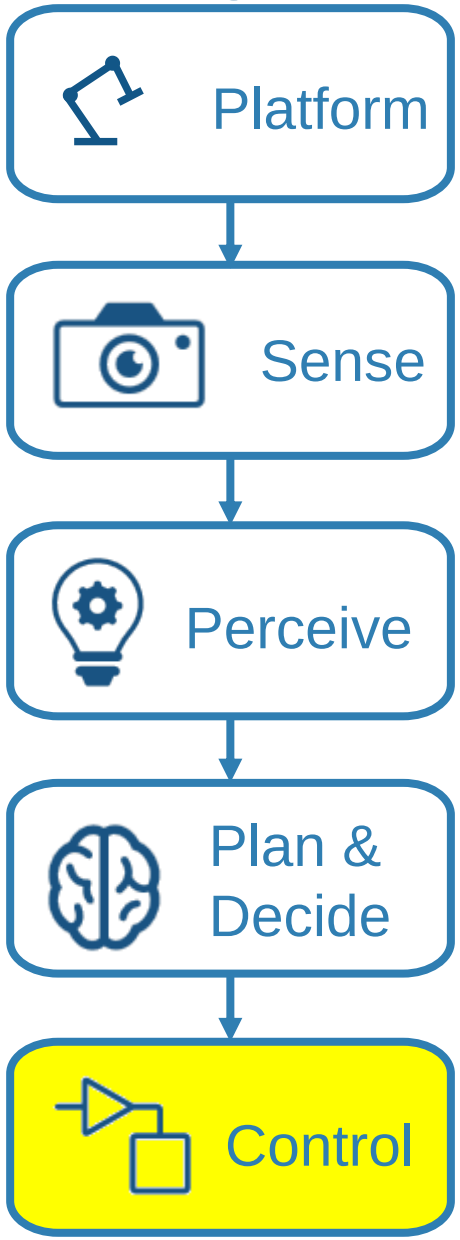
Planning: Find a path

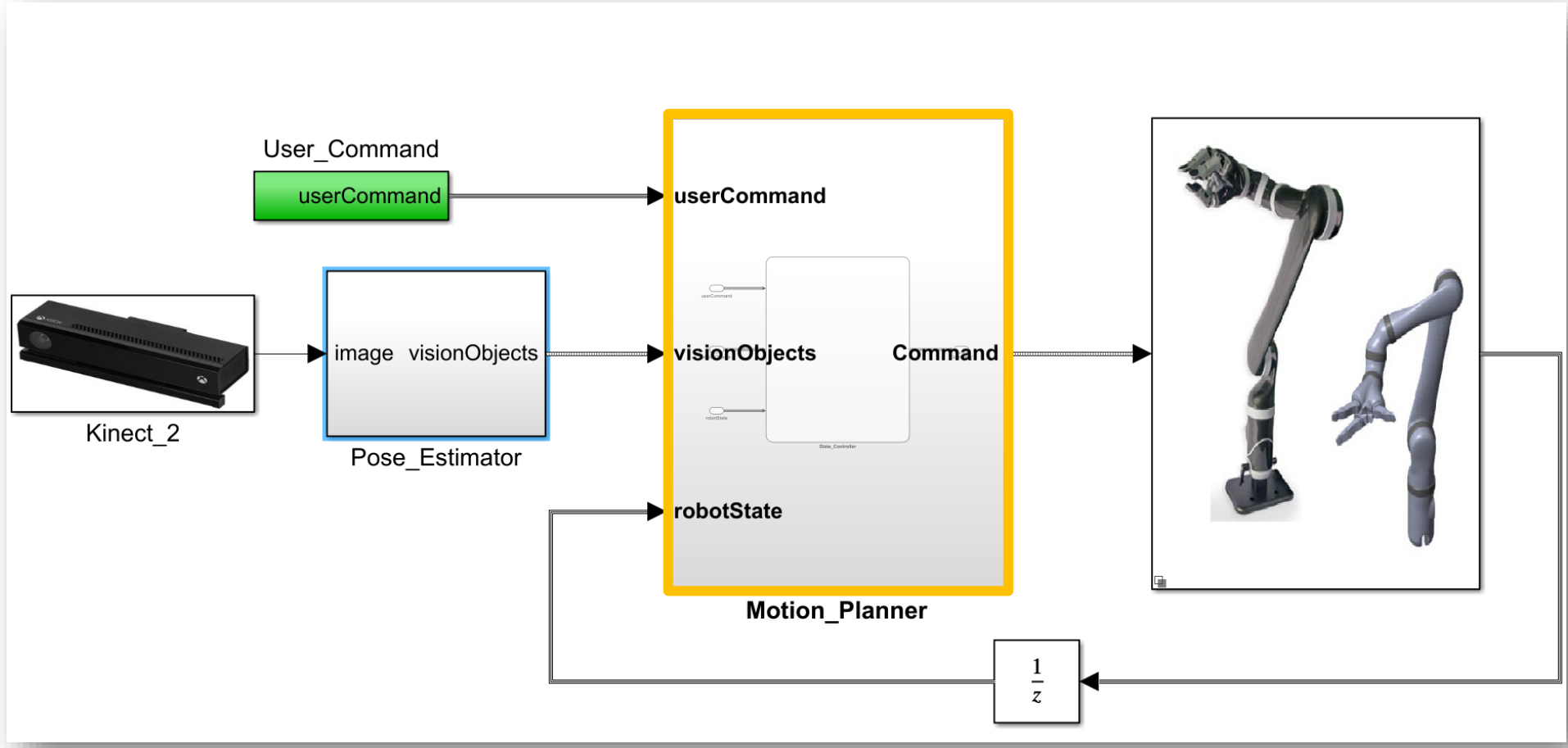


Plan with Stateflow



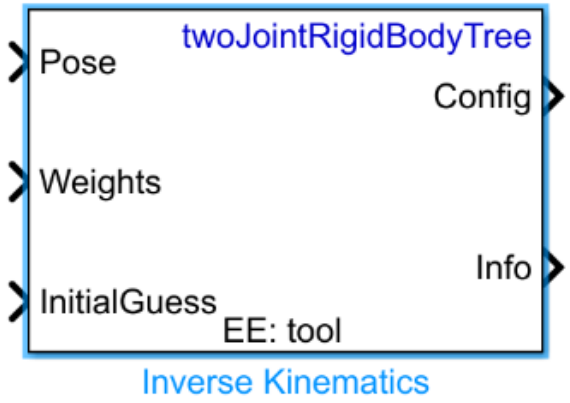
Design Pick and Place Application

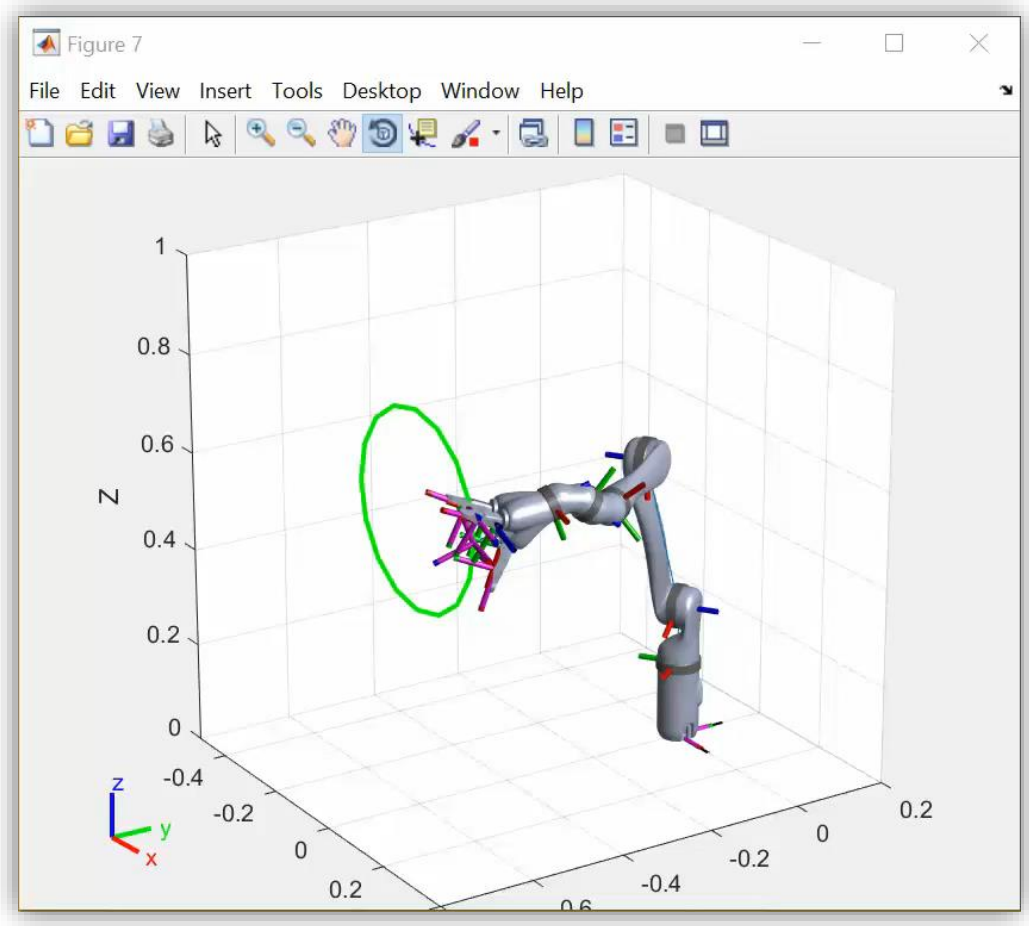




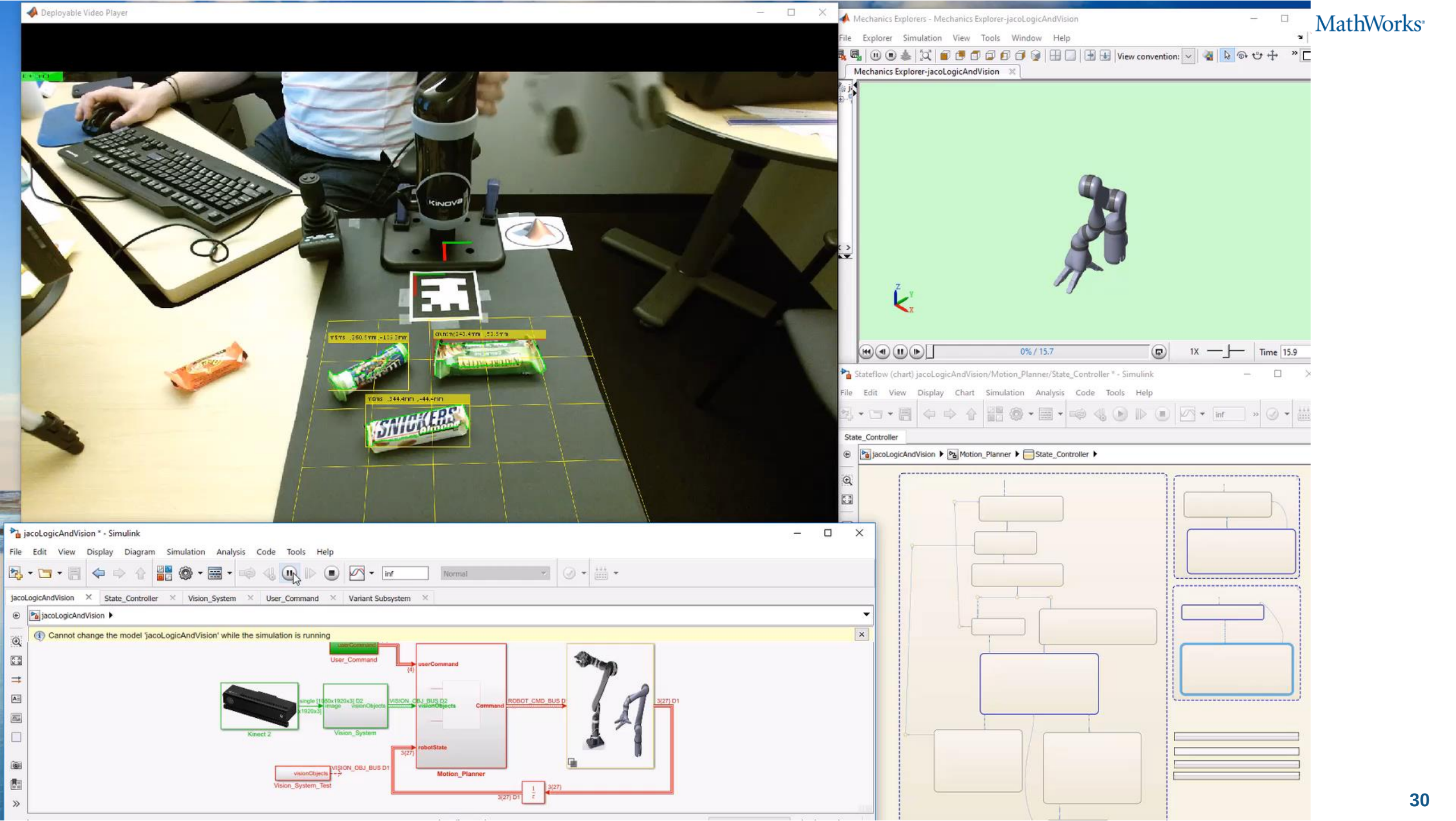
Control: Explore Built In Functions: Inverse Kinematics

```
% Create ik solver object
ik=robotics.InverseKinematics('RigidBodyTree',
                              jaco)

% Disable random restarts
ik.SolverParameters.AllowRandomRestart = false;

% Parameters to pass to the solver
weights = [1, 1, 1, 1, 1, 1];
q_init = 0.1*ones(numel(q_home),1);
```





Demo


Deployable Video Player
Mechanics Explorers - Mechanics Explorer-jacoLogicAndVision
File Explorer Simulation View Tools Window Help
View convention:
Mechanics Explorer-jacoLogicAndVision
0% / 15.7
1X
Time 15.9
Stateflow (chart) jacoLogicAndVision/Motion_Planner/State_Controller * - Simulink
File Edit View Display Chart Simulation Analysis Code Tools Help
State_Controller
jacoLogicAndVision
Motion_Planner
State_Controller
jacoLogicAndVision * - Simulink
File Edit View Display Diagram Simulation Analysis Code Tools Help
jacoLogicAndVision
State_Controller
Vision_System
User_Command
Variant Subsystem
Cannot change the model 'jacoLogicAndVision' while the simulation is running
User_Command
userCommand
Kinect 2
Vision_System
VISION_OBJ_BUS D2
visionObjects
Command
ROBOT_CMD_BUS D
3(27) D1
robotState
3(27)
visionObjects
VISION_OBJ_BUS D1
Vision_System_Test
Motion_Planner
3(27) D1
3(27)

Key Takeaway of this Talk

Success in developing an autonomous robotics system requires:

- Multi-domain simulation
- Trusted tools which make complex workflows easy and integrate with other tools
- Model-based design

Clearpath Robotics Accelerates Algorithm Development for Industrial Robots

Challenge

Shorten development times for laser-based perception, computer vision, fleet management, and control algorithms used in industrial robots

Solution

Use MATLAB to analyze and visualize ROS data, prototype algorithms, and apply the latest advances in robotics research

Results

- Data analysis time cut by up to 50%
- Customer communication improved
- Cutting-edge SDV algorithms quickly incorporated

An OTTO self-driving vehicle from Clearpath Robotics.

"ROS is good for robotics research and development, but not for data analysis. MATLAB, on the other hand, is not only a data analysis tool, it's a data visualization and hardware interface tool as well, so it's an excellent complement to ROS in many ways."

- Ilia Baranov, Clearpath Robotics

Link to user story

MathWorks® | Training Services

Deep Learning with MATLAB

This two-day course provides a comprehensive introduction to practical deep learning using MATLAB®.

Topics include:

- Importing image and sequence data
- Using convolutional neural networks for image classification, regression, and object detection
- Using long short-term memory networks for sequence classification and forecasting
- Modifying common network architectures to solve custom problems
- Improving the performance of a network by modifying training options

```
% Thank you
```