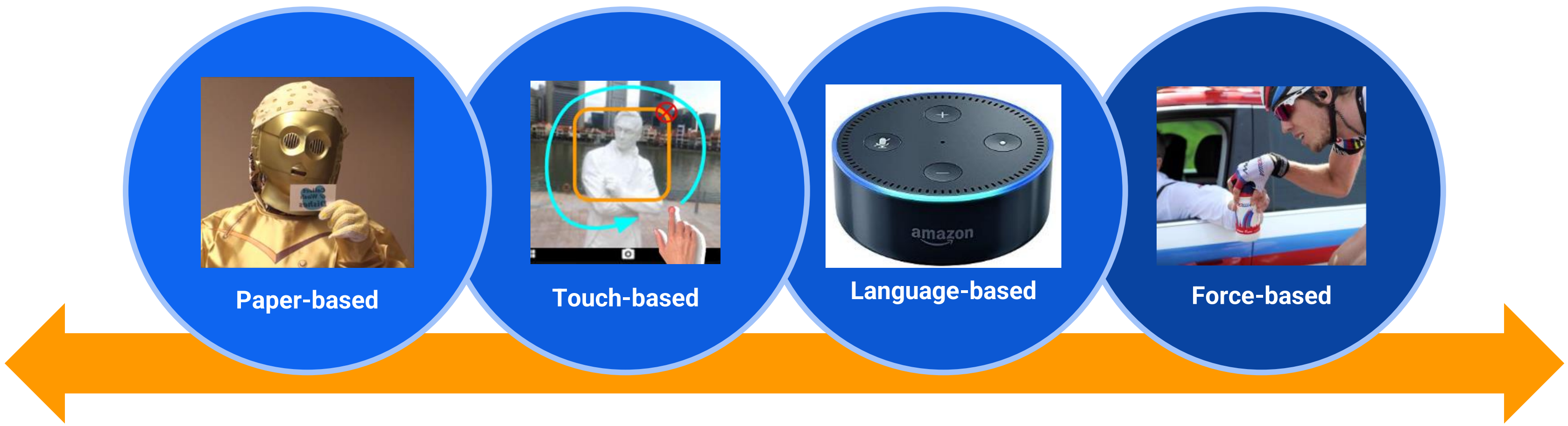
MODULE 6: HETEROGENEOUS HUMAN ROBOT INTERACTIVE SYSTEMS

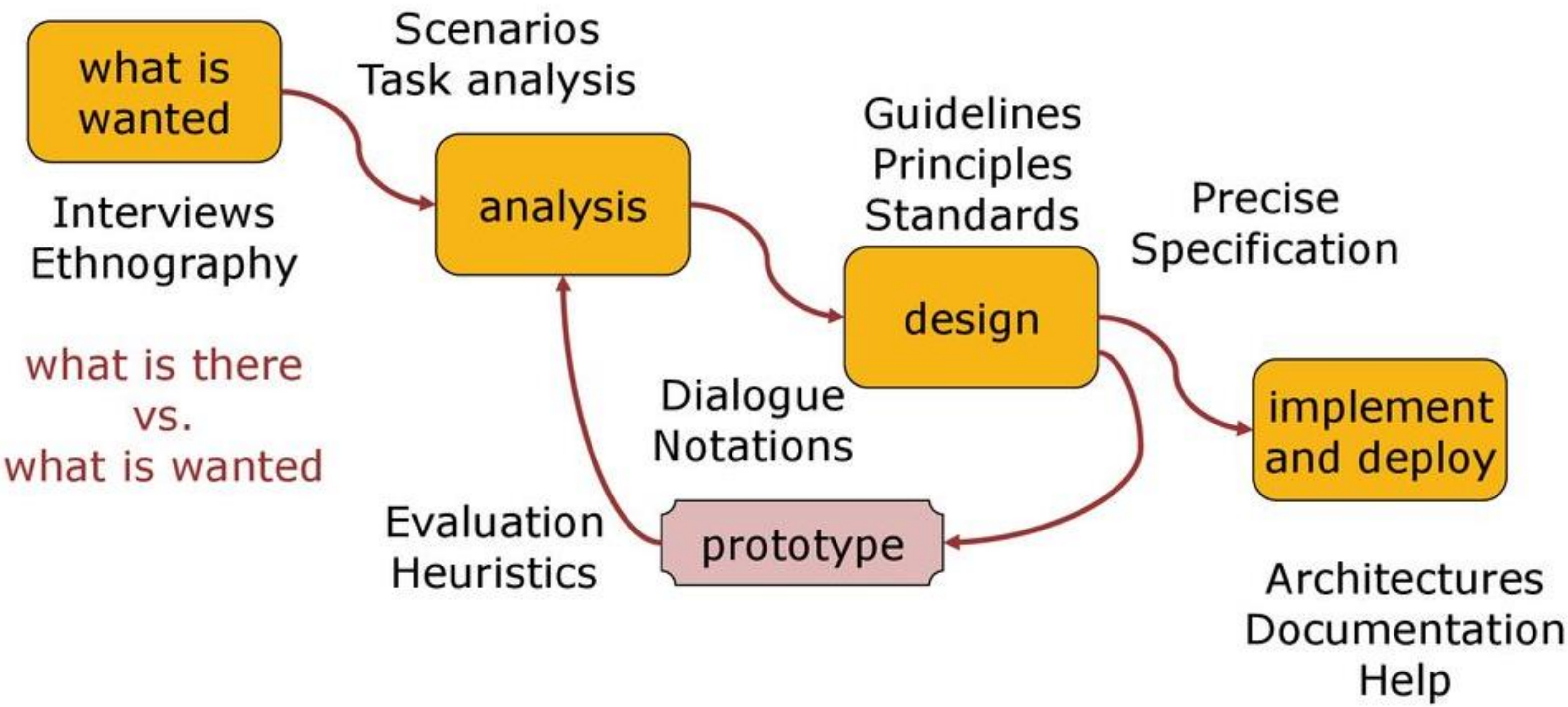
Nicholas Ho, PhD
Institute of System Science, NUS



Human
Factor

Robotics

Ideal Process



PAPER-BASED INTERFACE

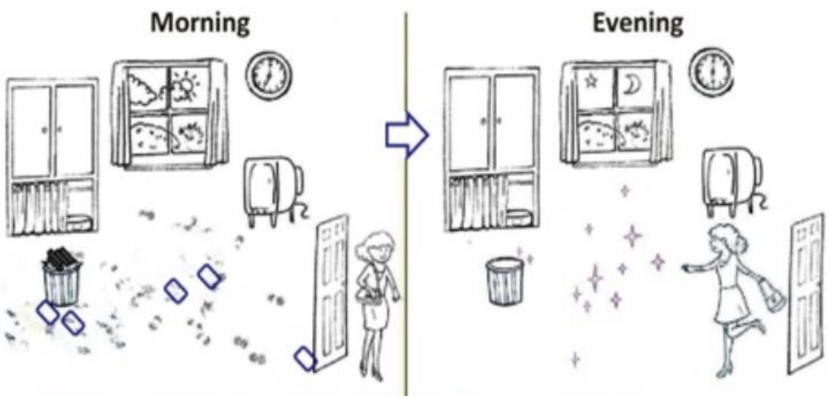
DOMESTIC HELPER EXAMPLE

1. What is Wanted

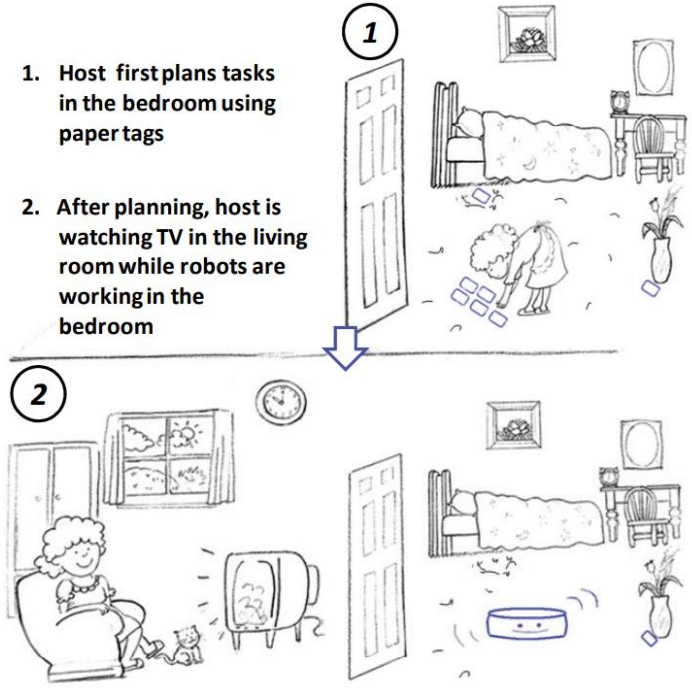
To have domestic housework done "magically", with the help of domestic-service robots

- **What is there? What is wanted?**
- **National Survey of Families and Households, etc**
- **Semi-structured interview**
  - 6 participants (1 male, 5 females, age 21-64, 3 married, 1 with children)
  - 2 hours per session
- **Outcome**
  - Learn living conditions and habits
  - Verify initial design ideas: paper cards, asynchronized interaction, etc

2. Analysis - Scenarios



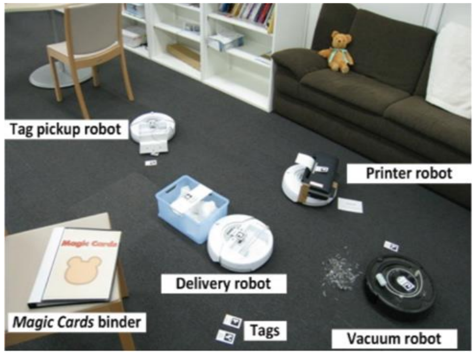
1. **In the morning, host planned tasks at home using paper tags, then is leaving for work**
2. **In the evening, host is coming back home and the tasks were completed**



3. Design

Achieving *goals*

within *constraints*



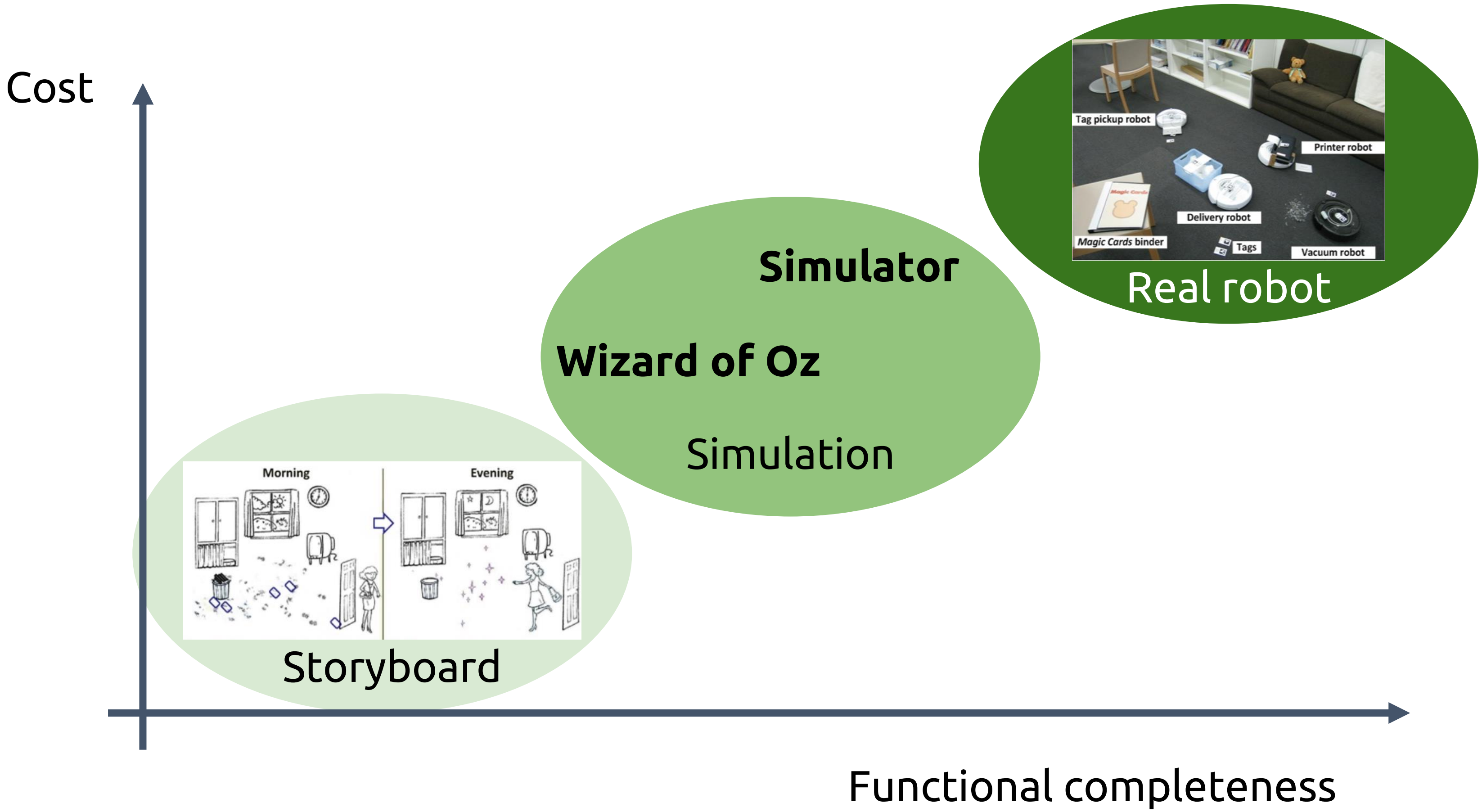
Goals:

- **What** is the purpose?
  - Do domestic housework
- **Who** is the design for?
  - E.g. working couples, housewives
- **Why** do they want?
  - Save trouble & time
- ...

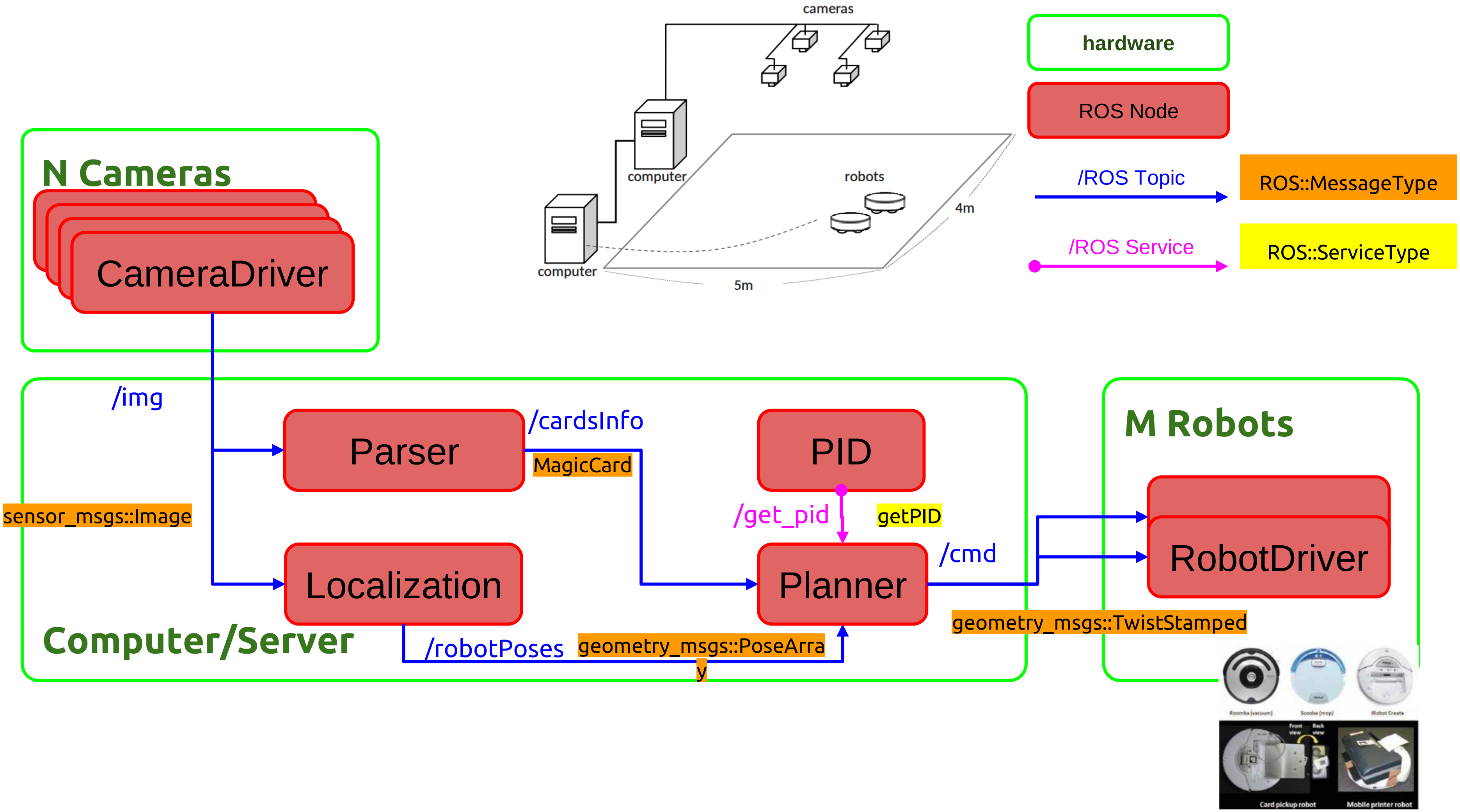
Constraints:

- Material
  - Paper cards, cameras, robots, servers
- Cost
  - $, $, $$$, $$
- Safety
  - Lost cards, broken robots, etc
- ...

4. Prototypes



5. Implement and Deploy



TOUCH INTERFACE DESIGN

PHOTOGRAPHER EXAMPLE

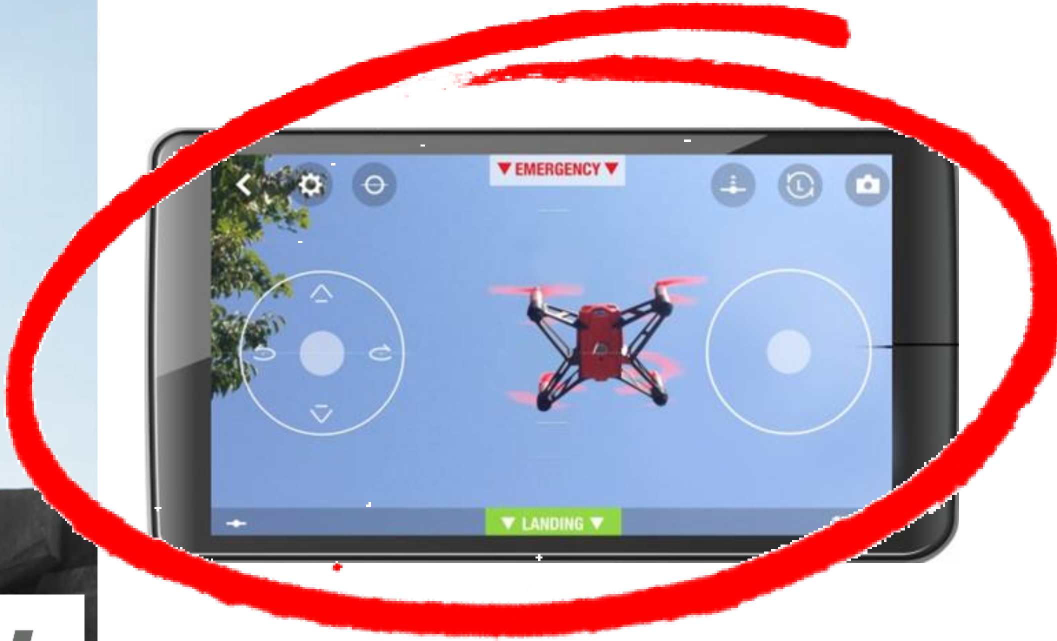
1. What is Wanted?

1. What is Wanted?

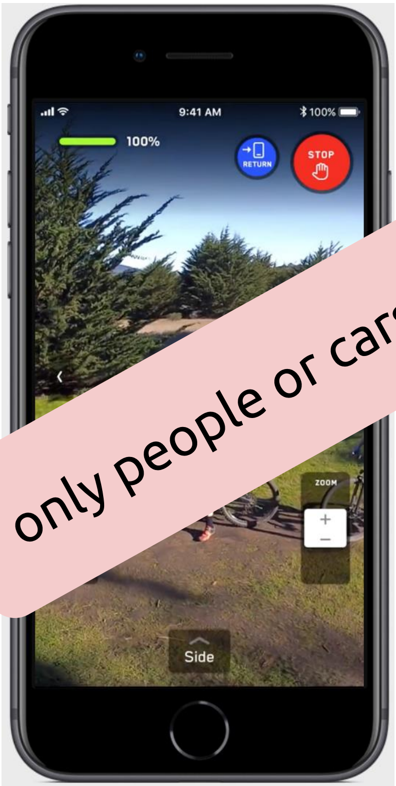
1. What is Wanted?

1. What is Wanted?

Flying camera interfaces

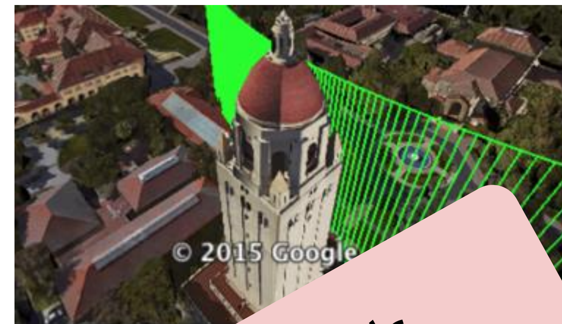


Various Features

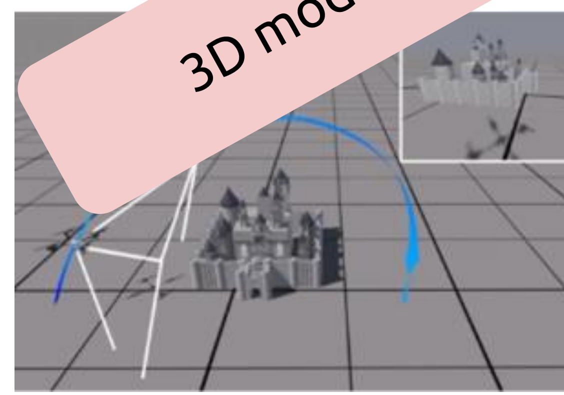


Skydio

Parrot



[1]



[2]

[1] Gebhardt, C., et al. “Airways: Optimization-based planning of quadrotor trajectories according to high-level user goals”. In *CHI 2016*
[2] Ross, S., et al. "Learning monocular reactive UAV control in cluttered natural environments." In *ICRA 2013*

Interview study

3 professional photographers

5 amature photographers

2 drone flyers/ instructors

point-of-view, composition, ...

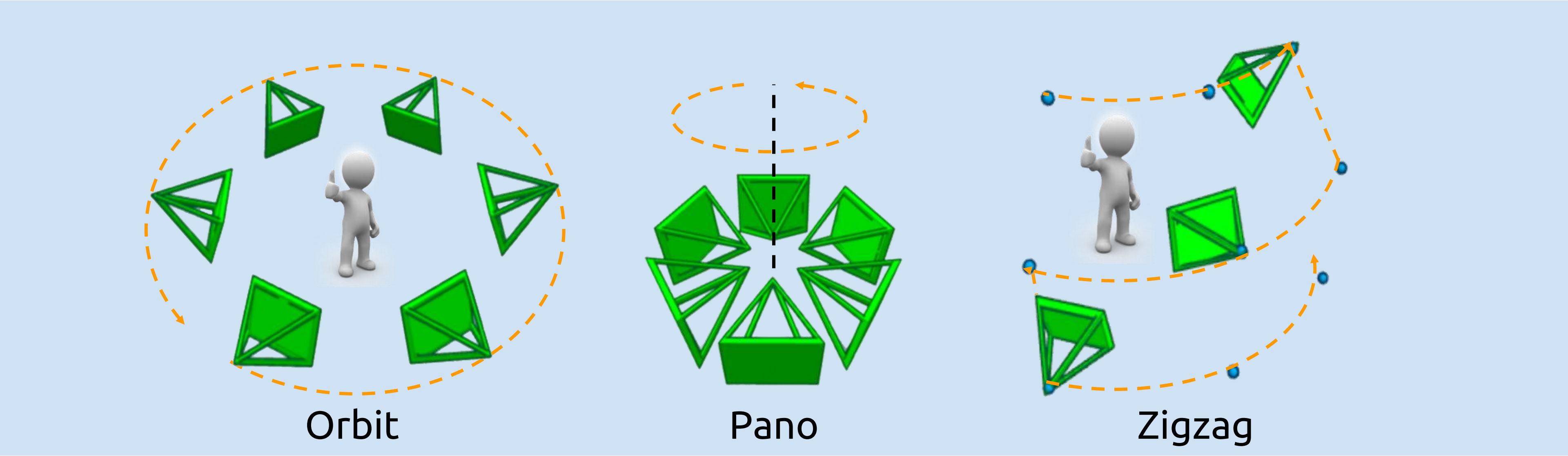
discover novel point-of-views, ...

extensive practice, tedious low-level control, ...

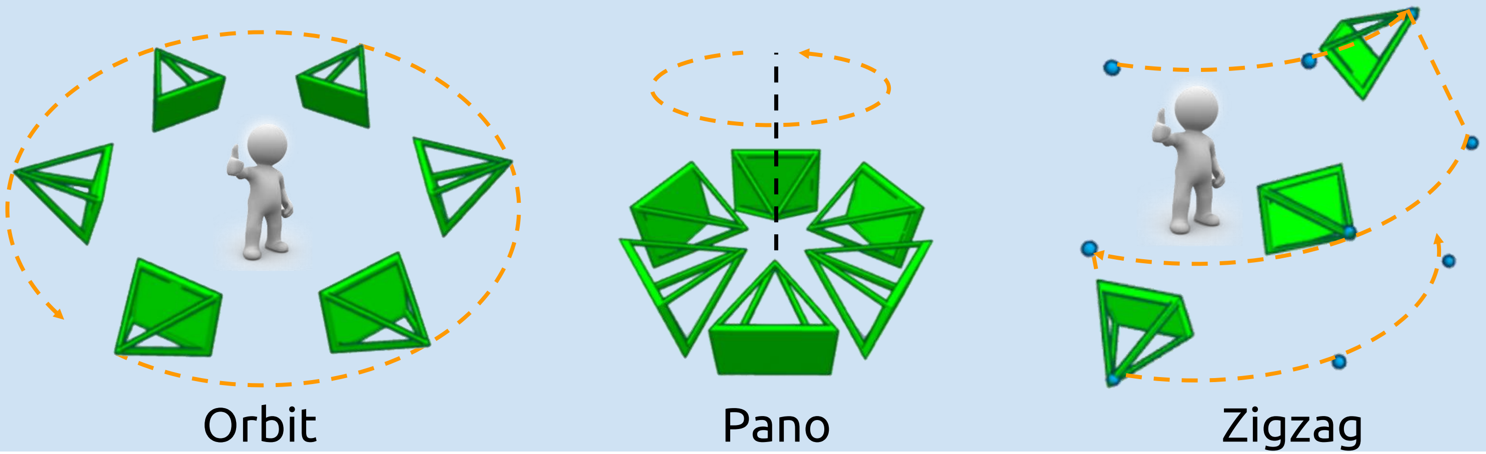
2. Analysis - Scenarios: Photo space

* images from the internet

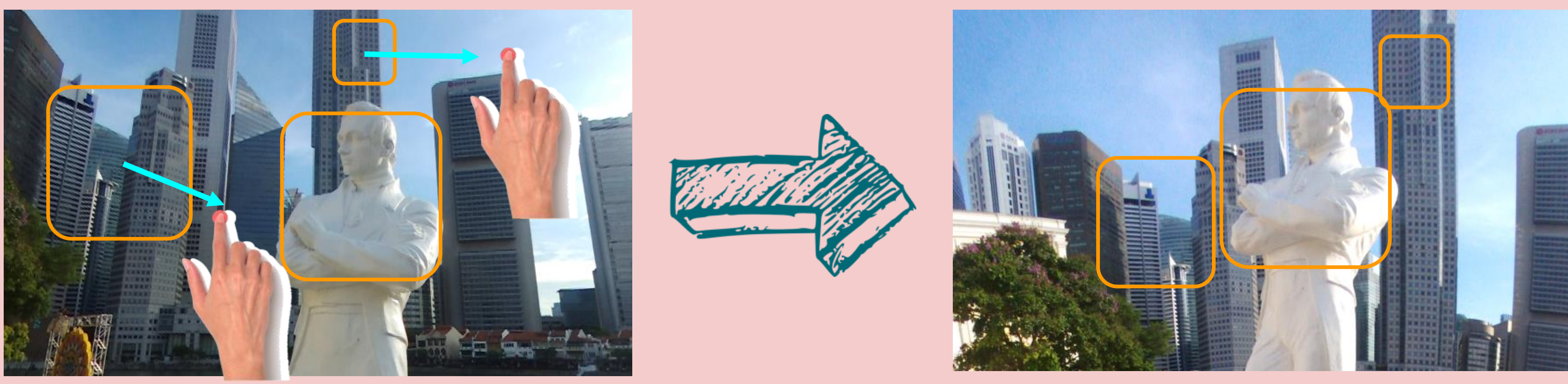
2. Analysis - Scenarios



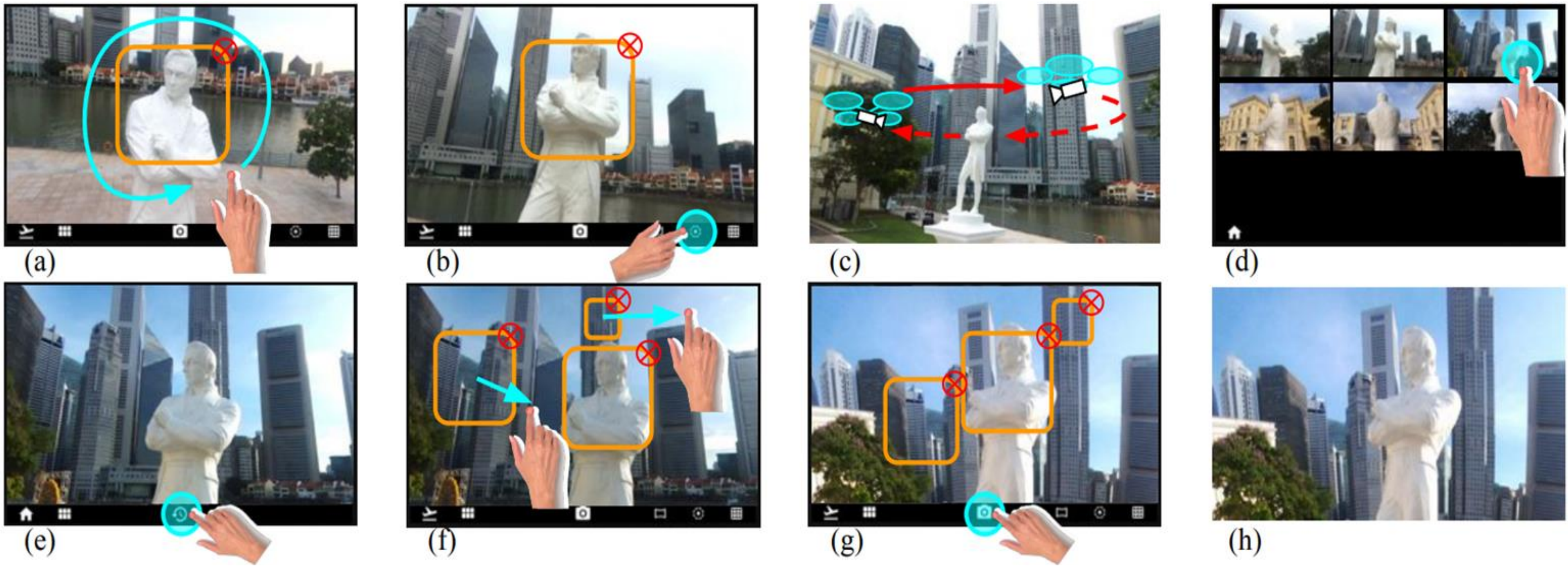
3. Design



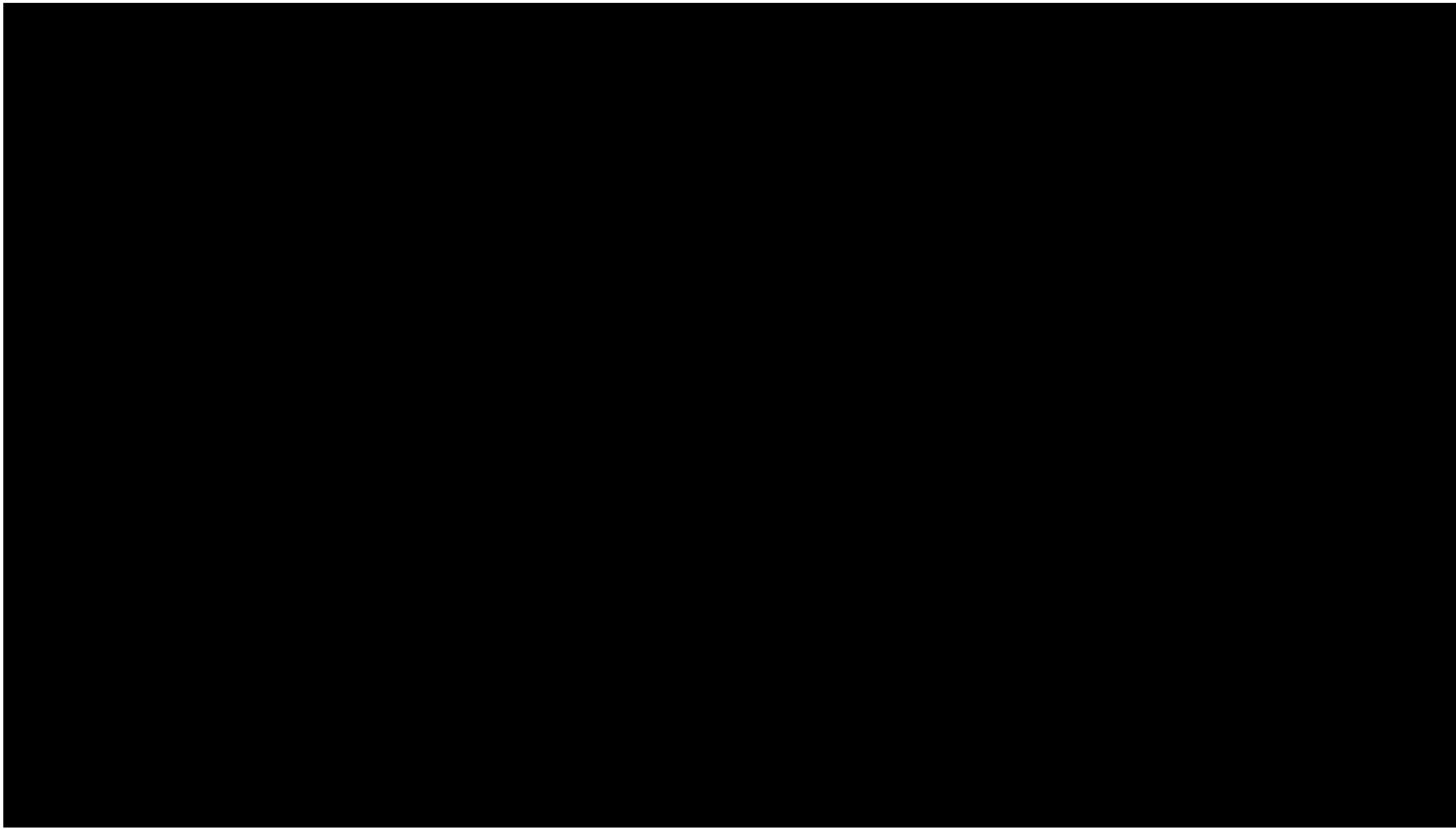
EXPLORE-AND-COMPOSE



4. Prototypes

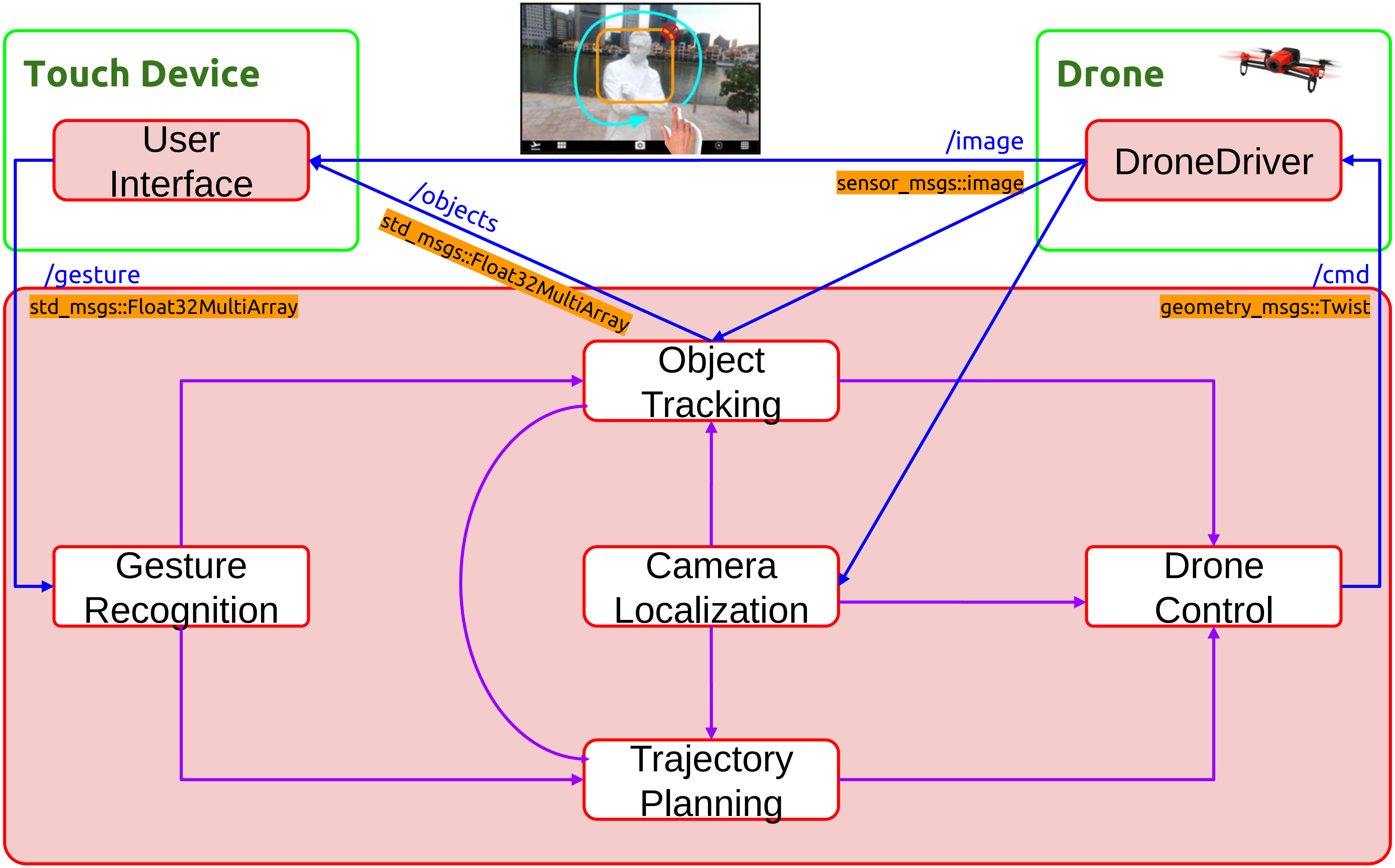


Touch Interface Example: Xpose (cont)



Source: https://www.youtube.com/watch?v=F1hrPb1SlHo

5. Implement and Deploy



VIDEO-LANGUAGE-MODEL (VLM)-BASED INTERFACE

BARTENDER EXAMPLE

1. What is Wanted???

2. Analysis - Scenarios

Simple command

Command with spatial relation

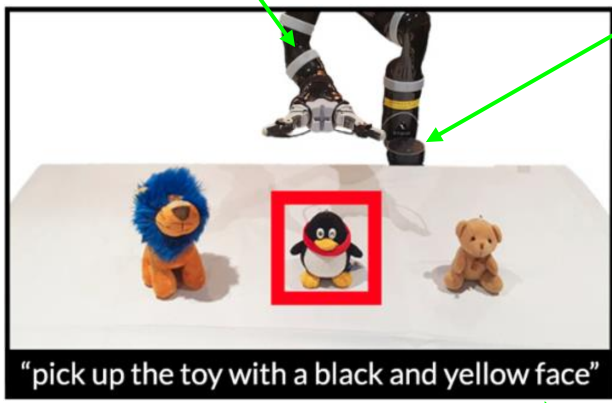
Q&A to disambiguate

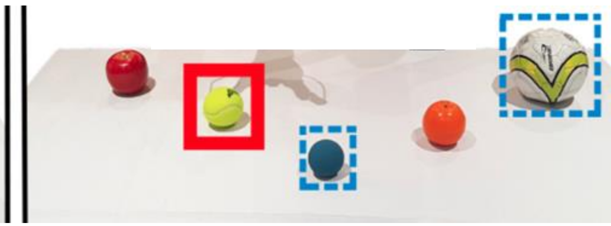
Use voice to command the robot to pick up a target object

3. Design

Mico robot arm

Amazon Echo





Simple command

verb. + an object with description

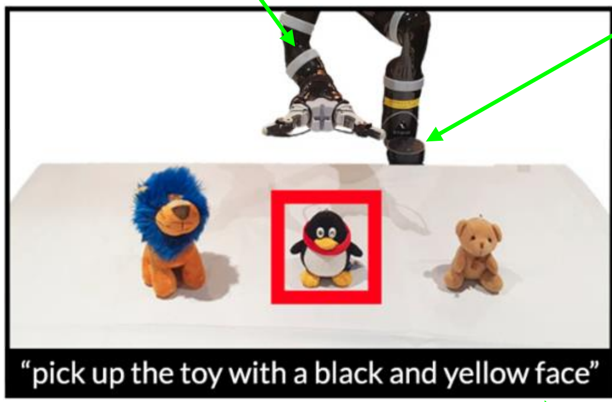
Command with spatial relation

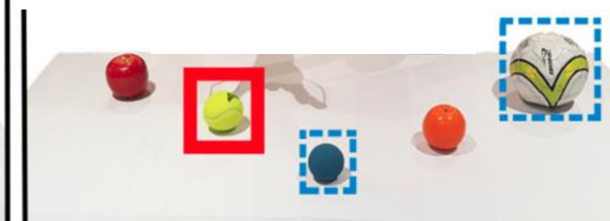
Q&A to disambiguate

3. Design

Mico robot arm

Amazon Echo





Simple command

Command with spatial relation

Q&A to disambiguate

verb. + an object with description

3. Design

Mico robot arm

Amazon Echo

Simple command

Command with spatial relation

Q&A to disambiguate

verb. + an object with description

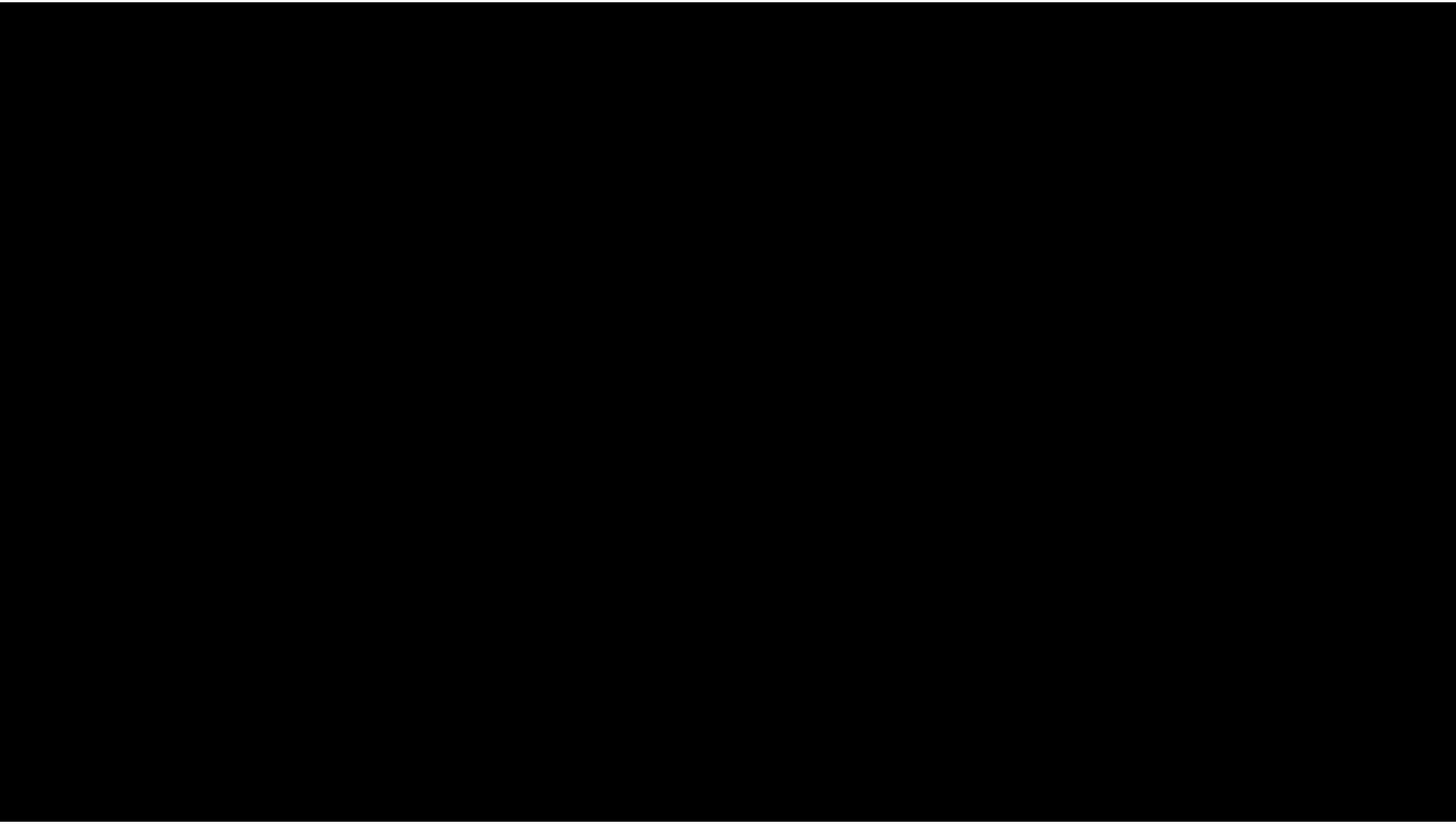
4. Prototype - Storyboard

Simple command

Command with spatial relation

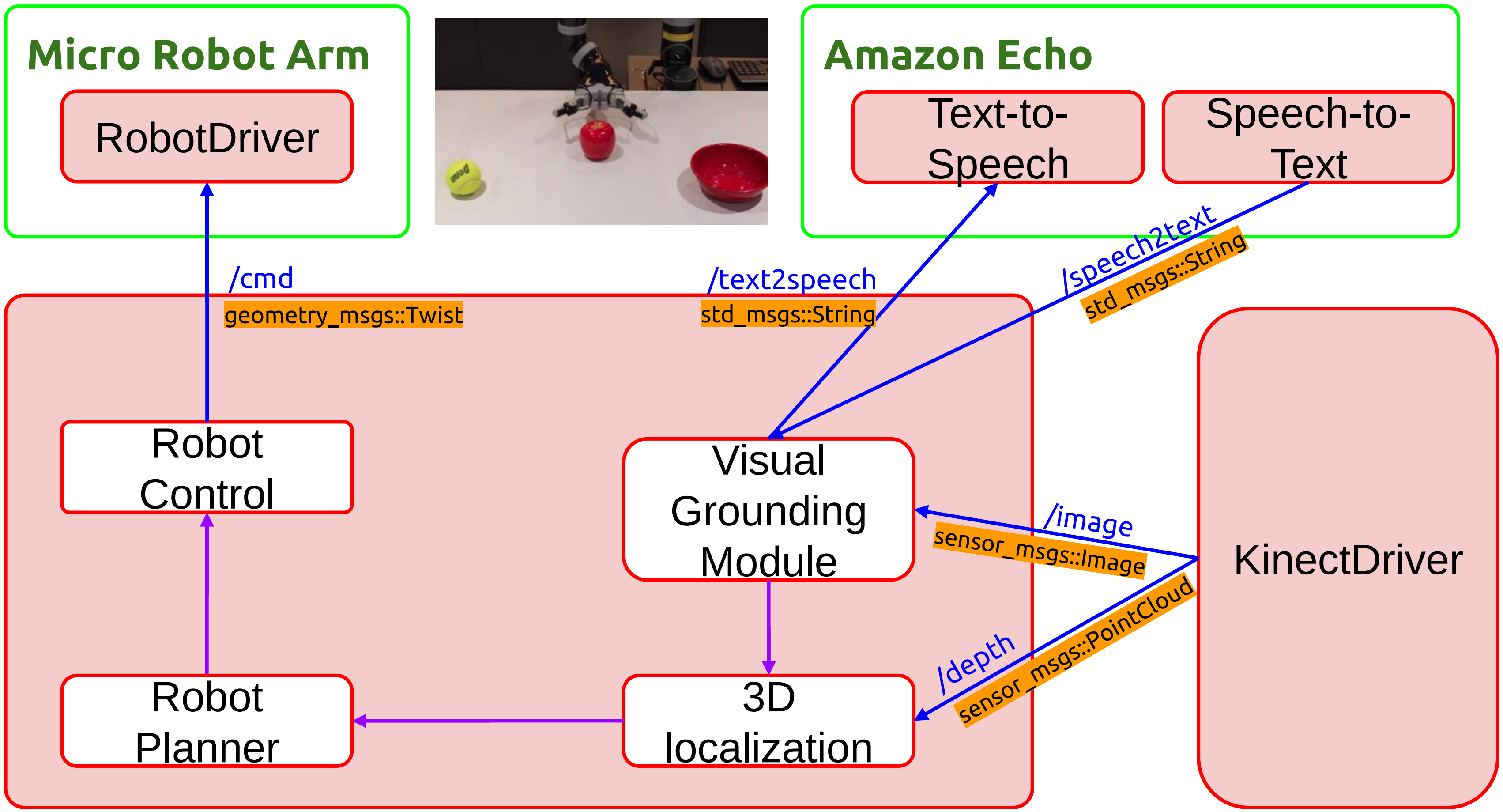
Q&A to disambiguate

4. Prototype - INGRESS



Source: https://drive.google.com/file/d/15AttCp-KCDEt8Ys5TfqXowsElm9GqAkH/view

5. Implement and Deploy



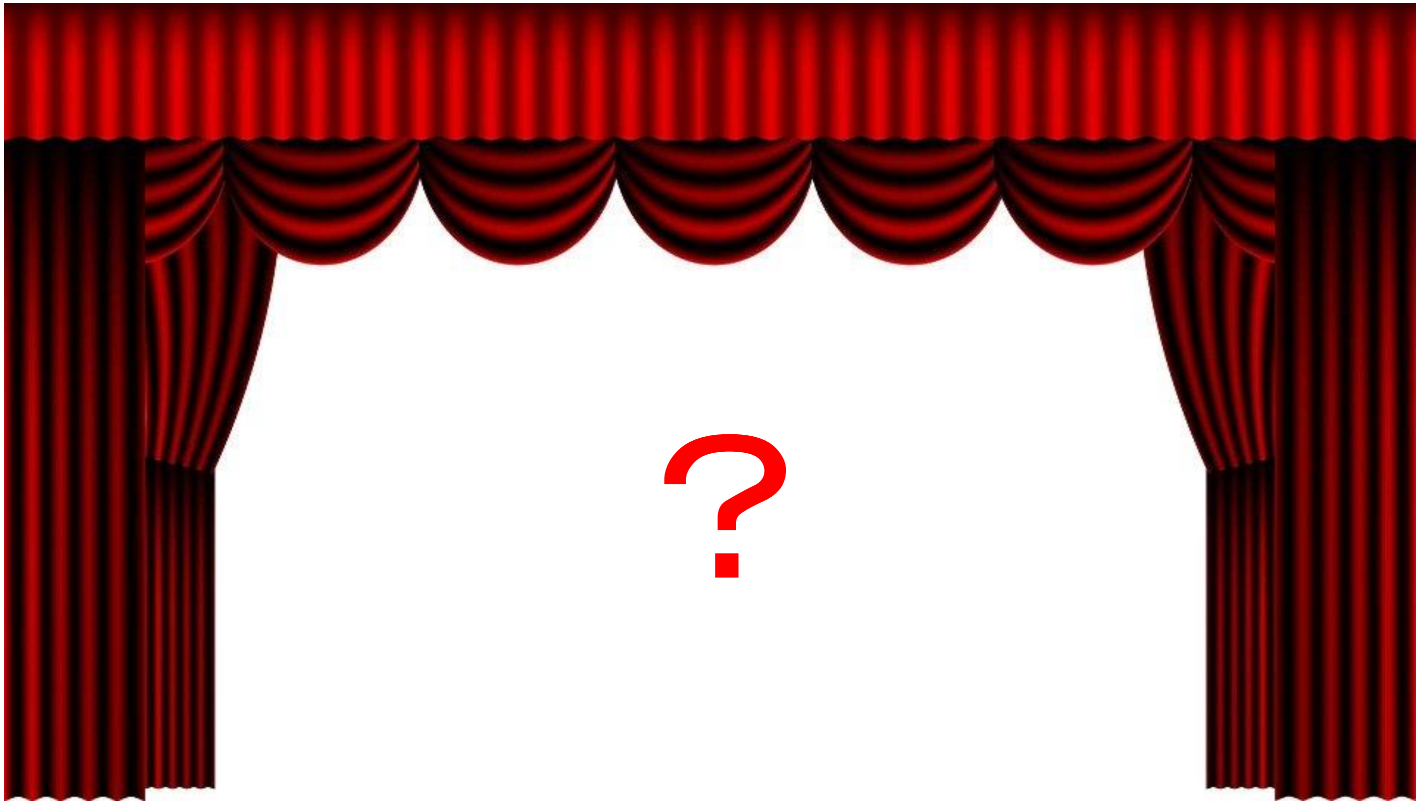
FORCE-BASED INTERFACE

HANDOVER EXAMPLE

1. What is Wanted

Sources:
https://www.youtube.com/watch?v=C9fdbYgHOYM
https://www.youtube.com/watch?v=n53nnkV4S74

State-of-the-art Handover Interactive Robotic System



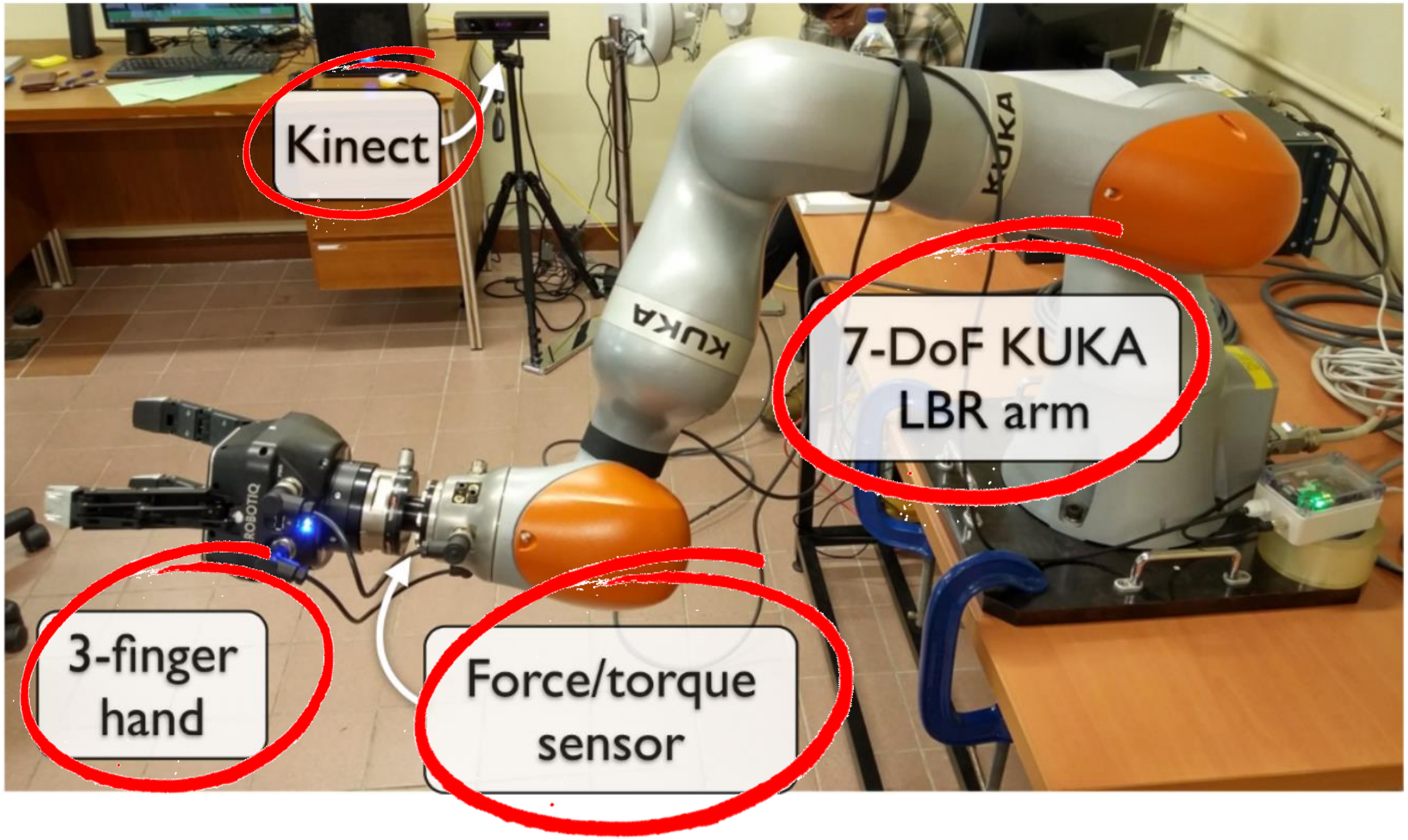
State-of-the-art Handover Interactive Robotic System

2. Analysis – Scenarios:
Handover while receiver is ...

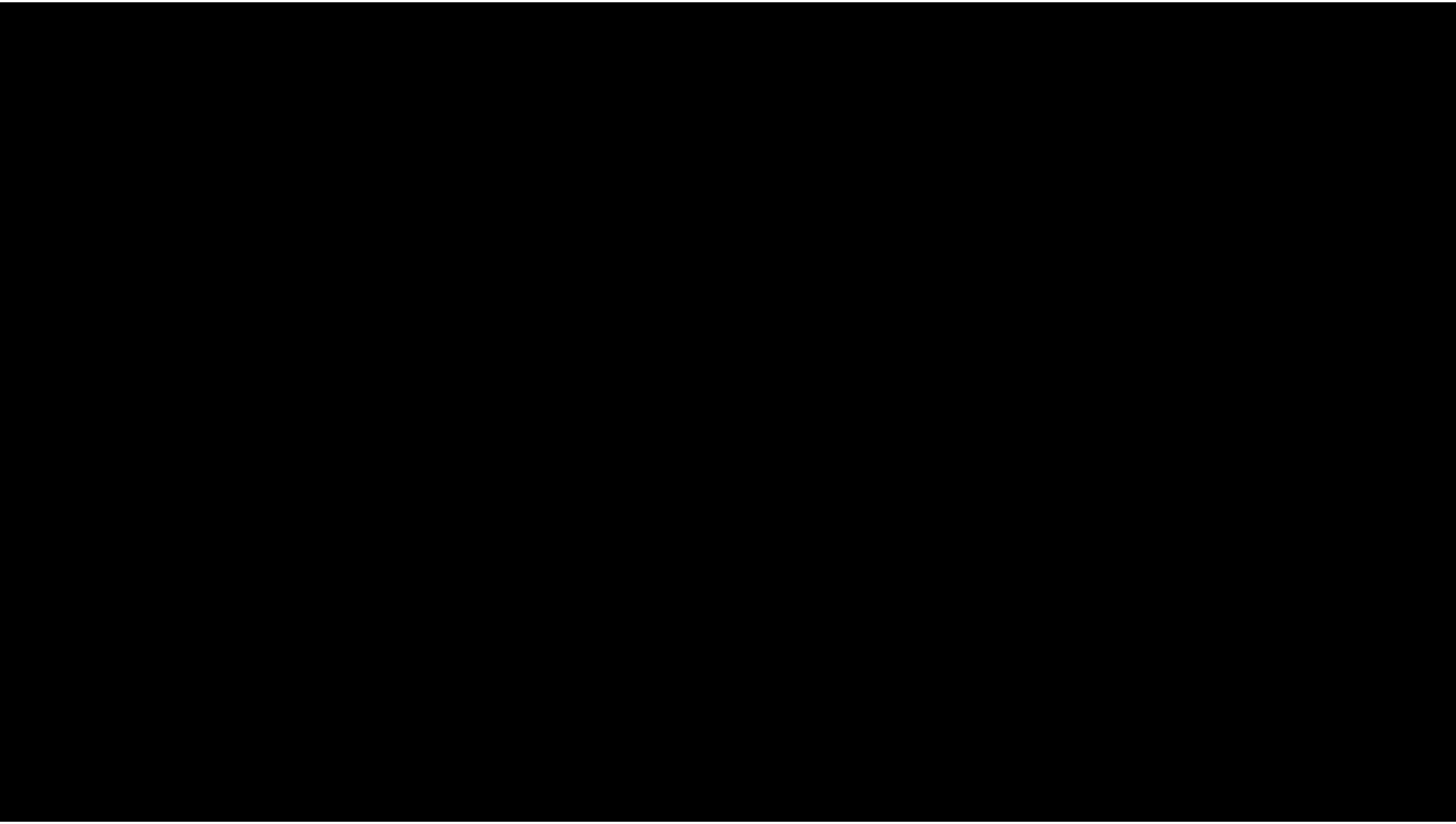
Standing *Walking* *Running* *Cycling*

3. Design


2ℓ
PURE
PACKAGED DRINKING WATER

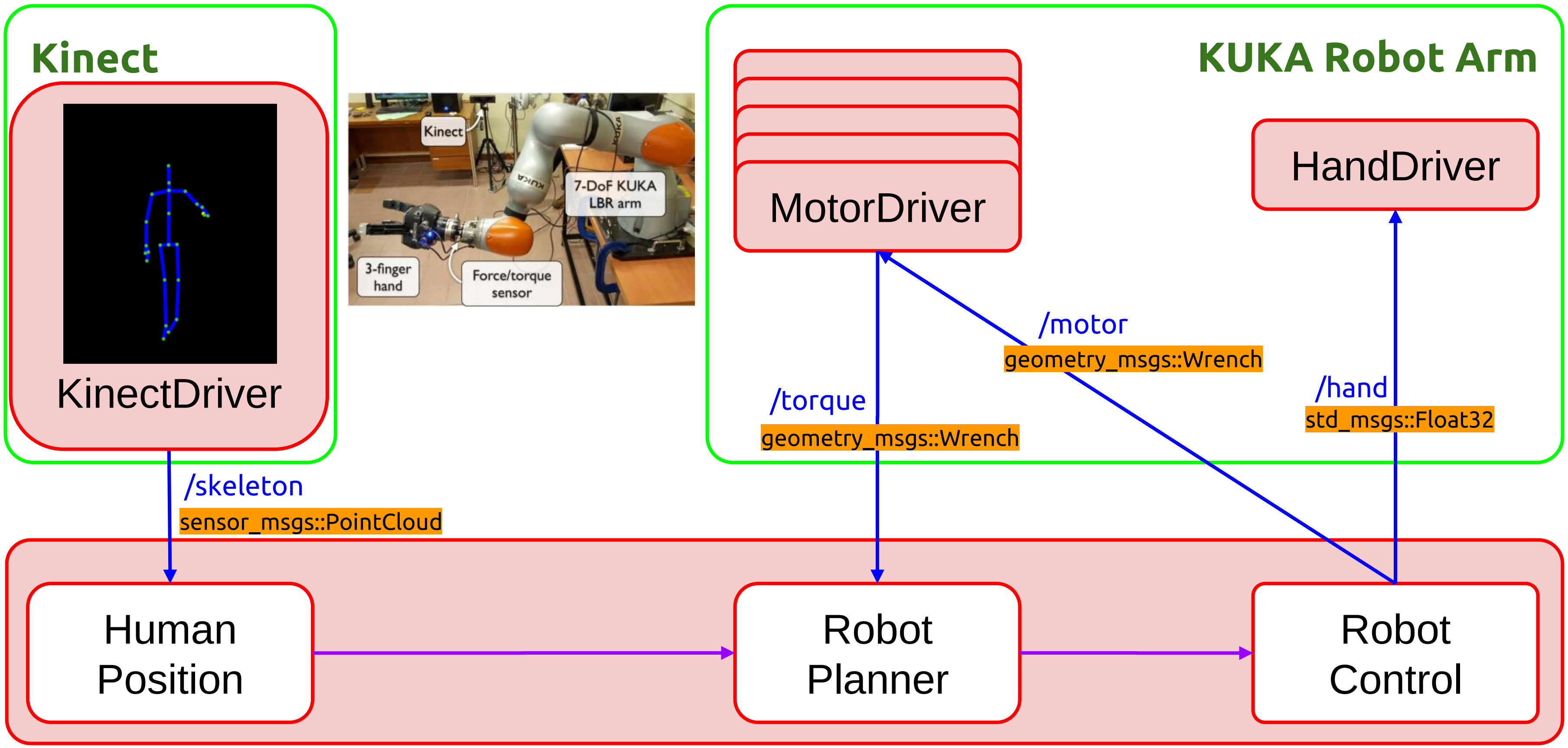

Kinect
7-DoF KUKA LBR arm
3-finger hand
Force/torque sensor
KUKA
KUKA

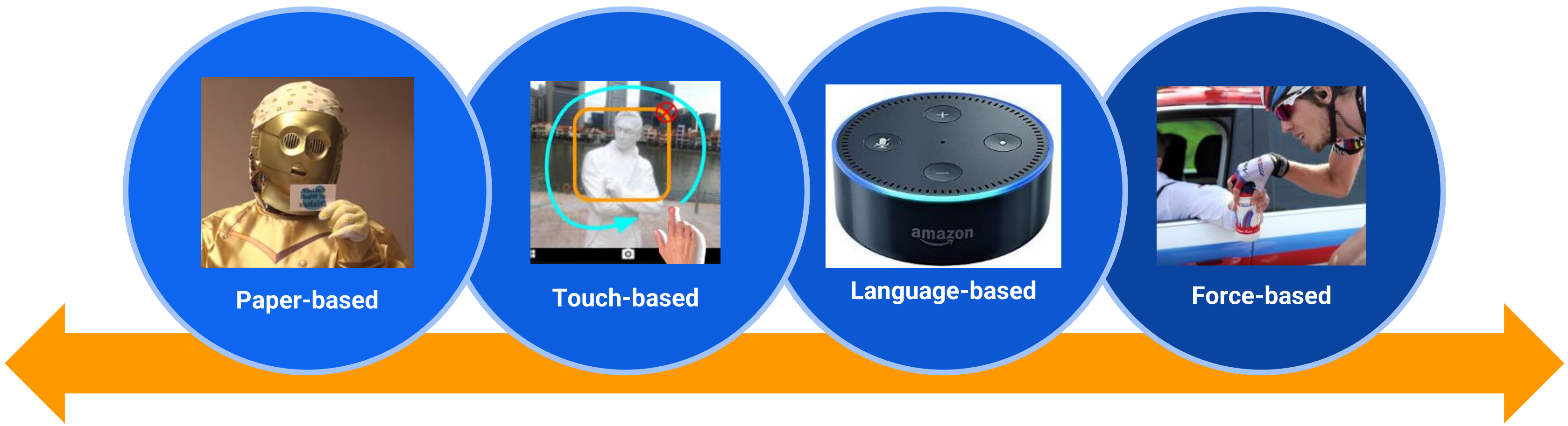
4. Prototypes



Source: https://www.youtube.com/watch?v=2OAnyfph3bQ

5. Implement and Deploy





Human Factor

Robotics

Others???

Vision-Based

Neural-Based

- Similar to MIT's neural-control robotic arm

Body-Language-Based

Less complex than the Disney's example

Gesture+Language-Based

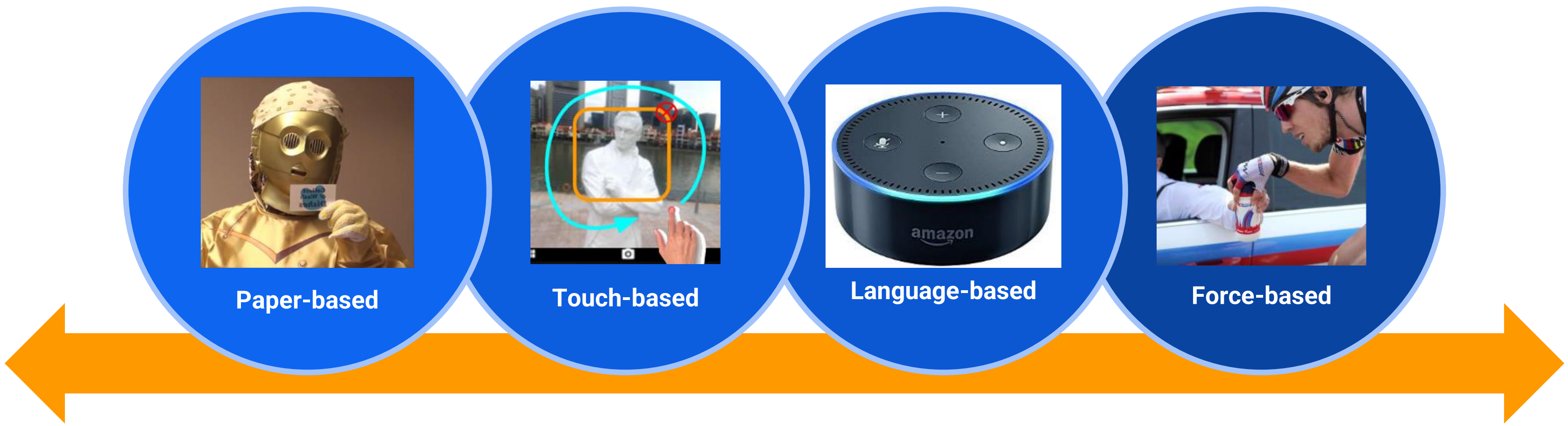
NUS National University of Singapore | Advanced Robotics Centre

Supported by:
Computing and Engineering

v1.9

Collaborative Intelligent Robotics System Based on KUKA IIWA

Chen Yiwen
cywgoog@gmail.com



Human Factor

Robotics

Generative AI-based

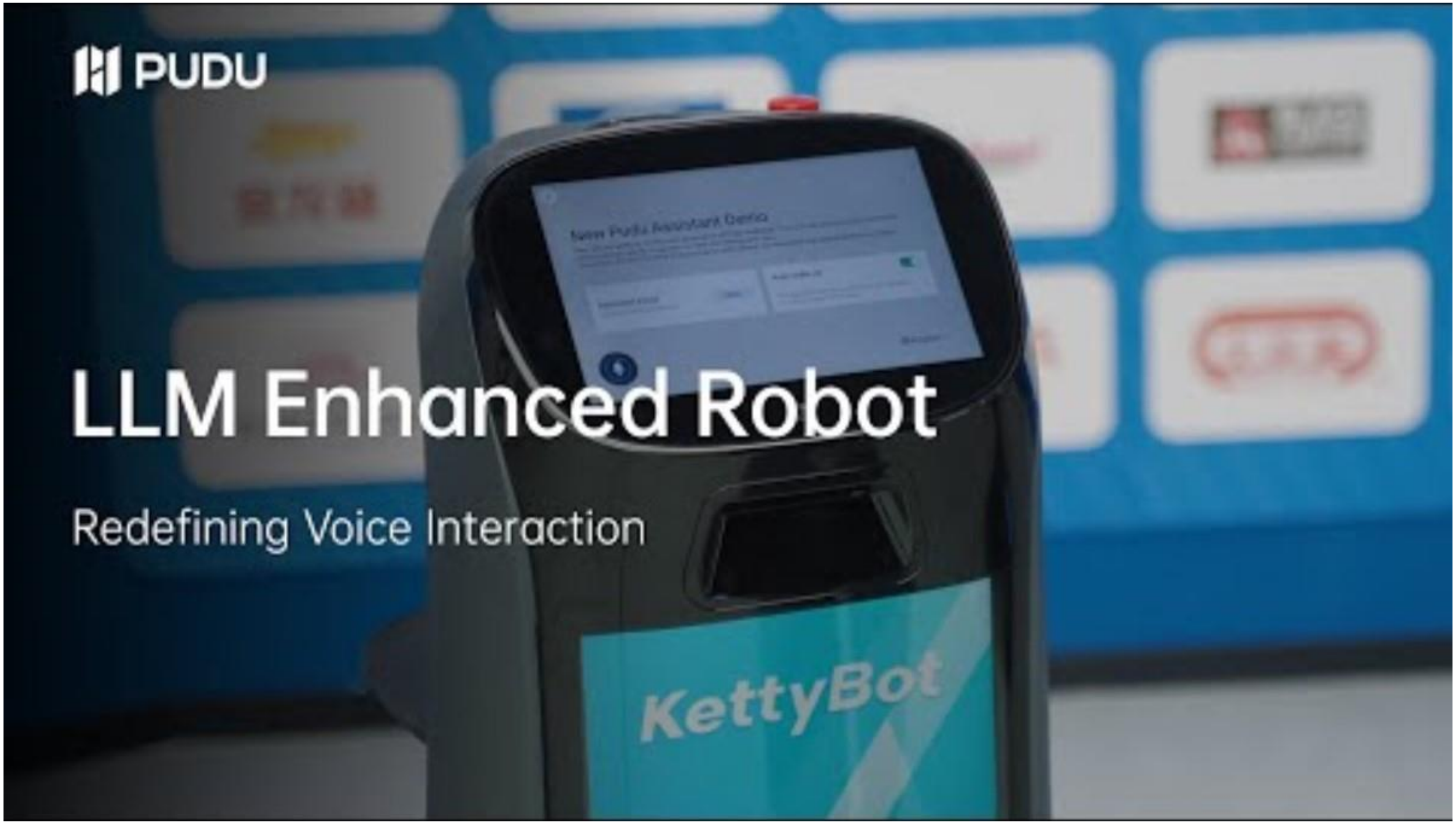
GenAI-based Robots

1. LLM-based Robots
   - E.g. A virtual assistant robot generating detailed answers to complex questions in real-time over speech.
2. VLLM-based Robots
   - E.g. A service robot in a retail store assisting customers by understanding visual cues (such as product images or customer gestures) and generating detailed, real-time responses over speech.
3. Creative Content Generation Robots
   - E.g. A robot that paints original paintings or plays original music based on generated data.
4. Multimodal Generative Robots
   - E.g. A healthcare robot that listens to a patient's verbal input, analyze their facial expressions and body language, and generate personalized care advice.

GenAI-based Robots

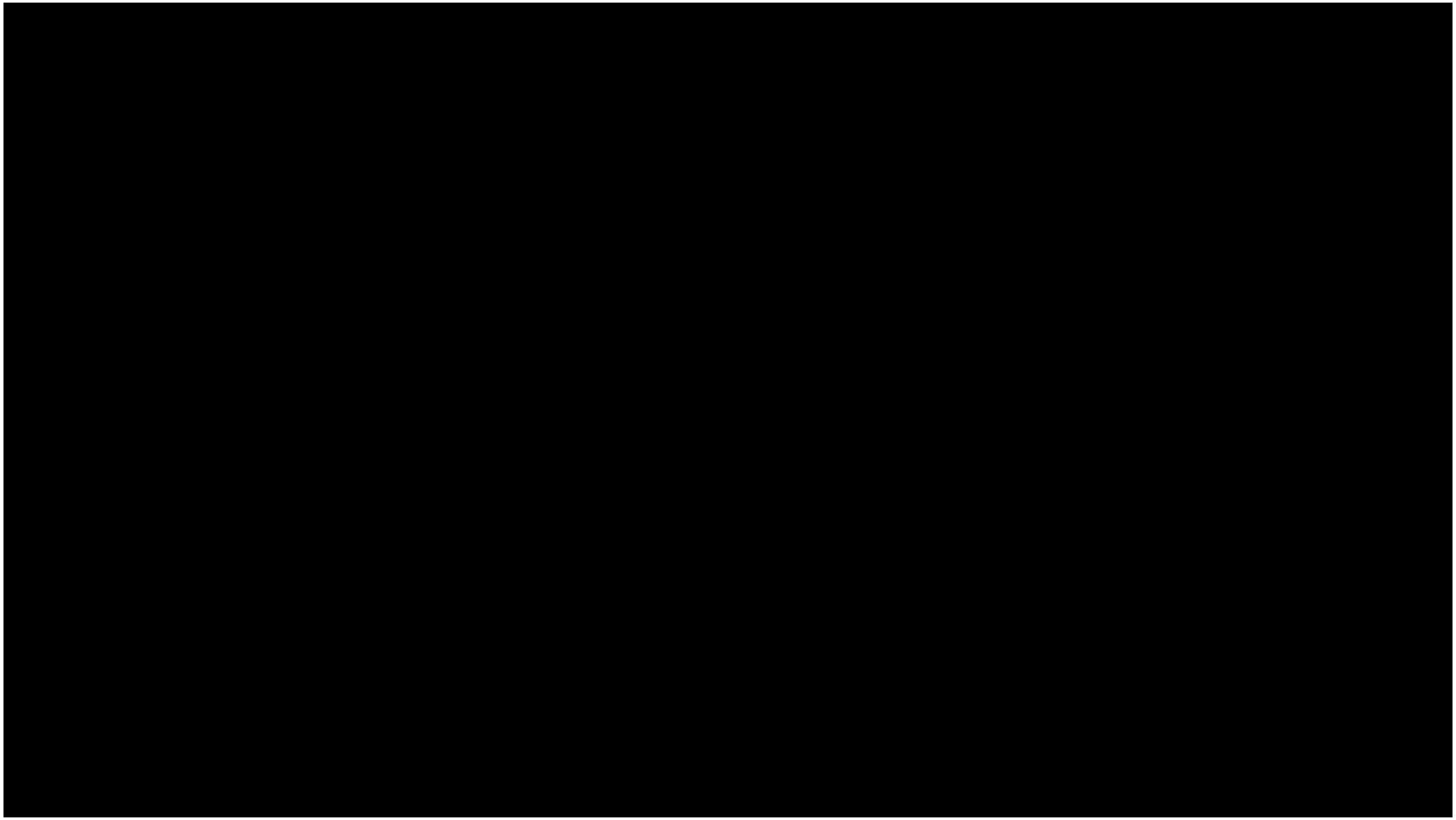
5. Emotions-Generated Robots
   - E.g. A social robot detecting sadness in a user and generates empathetic responses through dialogue and emotional cues to provide comfort or motivation.
6. Generative Planning and Decision-Making-based Robots
   - E.g. A logistics robot generating optimized delivery routes based on current traffic and package priorities, adjusting plans dynamically.
7. Generative Predictive Model-based Robots
   - E.g. A factory robot generating predictive maintenance models to predict equipment failures and generating preventative maintenance schedules.

(A) Large-Language-Models (LLM)-based robots: Pudu Robotics



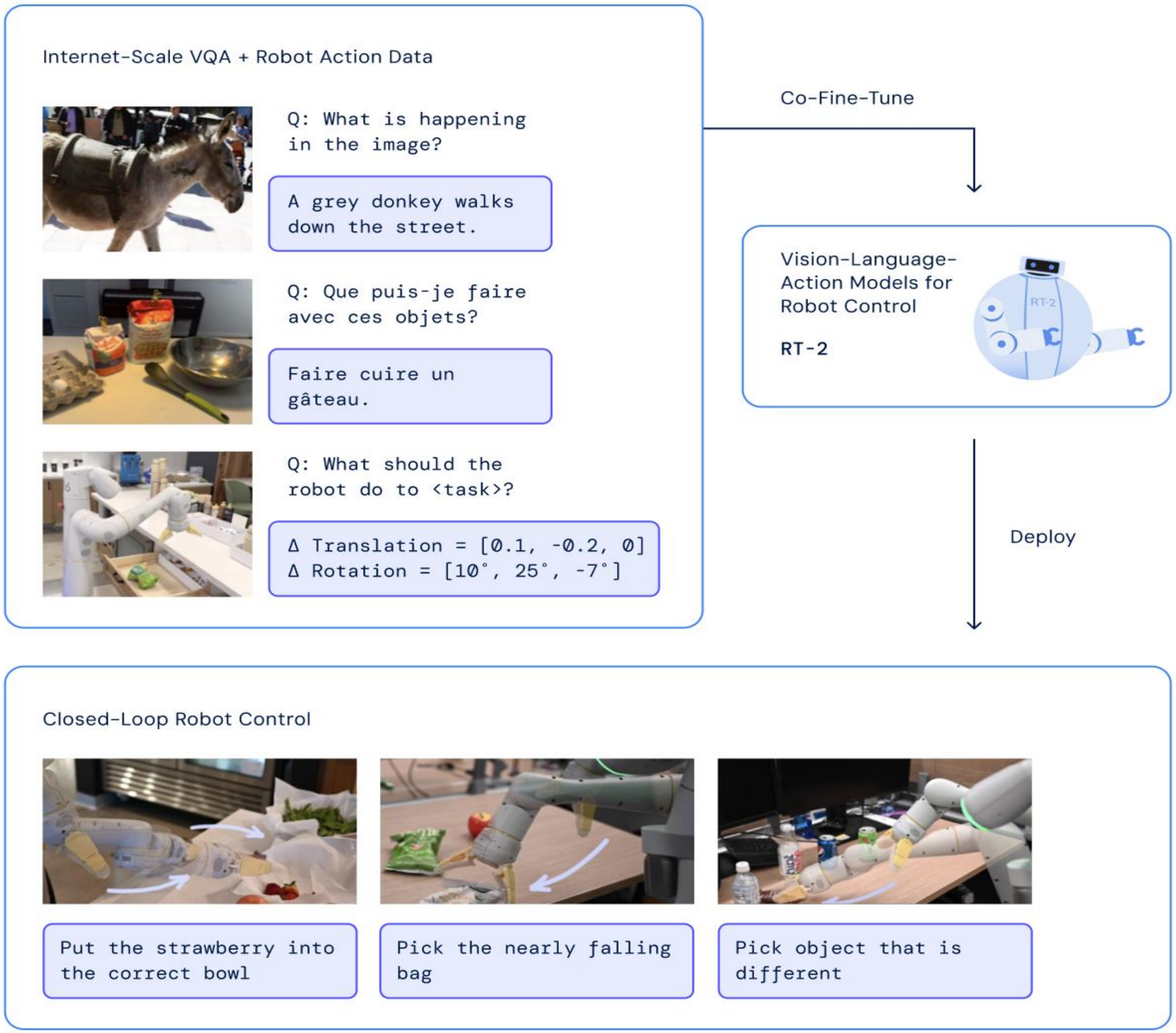
Source: https://www.youtube.com/watch?v=sh04-QB_AAU

(B) Vision-Large-Language-Models (VLLM)-based robots: “OK-Robot”



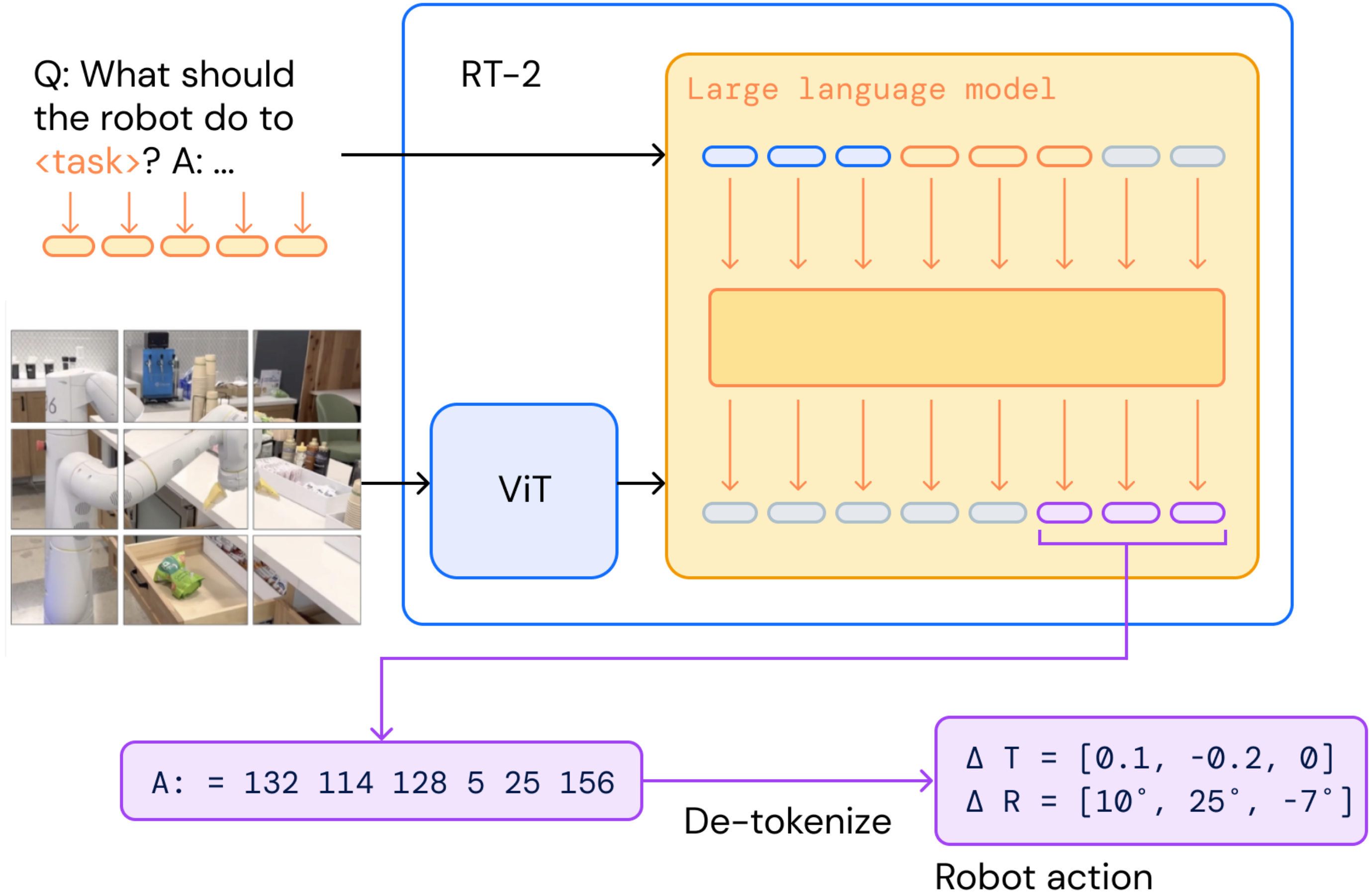
Source: https://www.youtube.com/watch?v=-zENt82QPEA

Another example of VLLM-based robots: “RT-2 Robot”



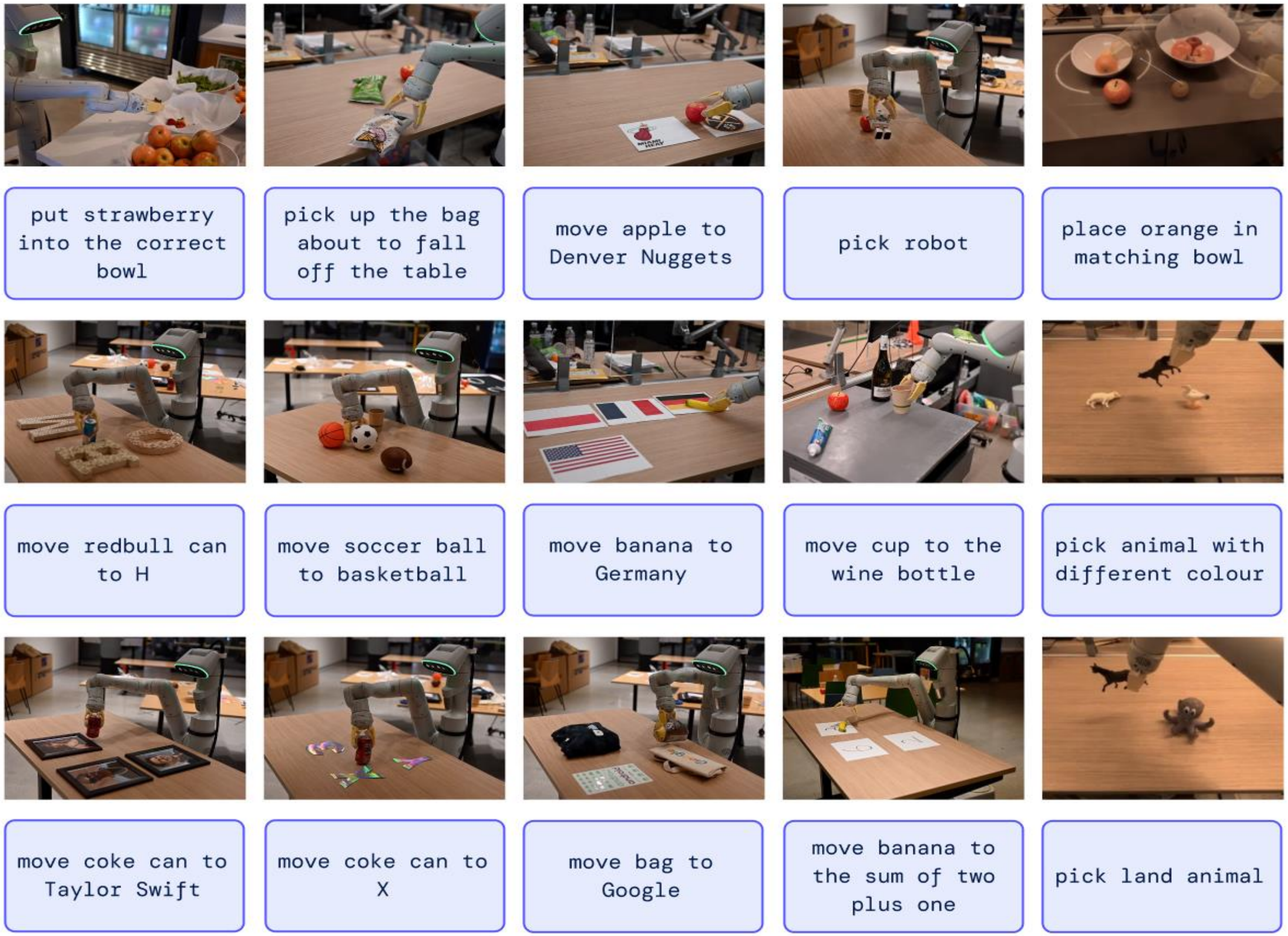
Source: https://robotics-transformer2.github.io/

Another example of VLLM-based robots: “RT-2 Robot”



Source: https://robotics-transformer2.github.io/

Another example of VLLM-based robots: “RT-2 Robot”



Source: https://robotics-transformer2.github.io/

Another example of VLLM-based robots: "RT-2 Robot"

Source: https://robotics-transformer2.github.io/

(C) MultiModal Gen AI-based robots: MD Robot

Source: https://www.youtube.com/watch?v=gumdtEr4bVg

MODULE 5 & 6 WORKSHOP: WORKSHOP DAY 3

WORKSHOP: HANDS-ON CONSTRUCTION OF MDP/HRIMDP MODELS OF AUTONOMOUS DRIVING

Personal Healthcare Robot

- Personal assistant robots gain their popularity, E.g., Google Home, Amazon Echo, Xiao Mi robot vacuum, ASUS Zenbo, etc
- They are perfect candidates to become the future personal healthcare givers that monitor the elderly users at the nursing home or in the hospital. In this workshop project, we formulate this interaction between the personal healthcare robot and the human user

Learning Objectives of Workshop Day 3

1. Create a planner program that plans and executes optimal robot movements with the presence of human
2. The HRIMDP method (covered in Workshop Day 2) will be utilized as the base to find the best policy
3. Simulation of random human actions within virtual world
4. Implementation of developed algorithms within various given virtual worlds (i.e. *world_maze*, *world_room*, *world_twist*)
5. Observation of various interaction patterns between human and robot within these worlds

Part A: Applying HRIMDP method in Robot Path Planning

Things to do

1. Complete function **robot_execute_under_policy** in **planner.py**
2. Test a complete run using the given human action file (i.e. human_a.csv)

   python3 human_execute.py world1.csv human_a.csv 2 3 1 human_s.csv

   python3 planner.py world1.csv human_s.csv 0 0 robot_s.csv

   python3 visualizer.py world1.csv human_s.csv robot_s.csv

3. Create 2 different variations of human action files and test the complete run using these 2 new human action files; save different trajectory files under these human action files. For example:

   human_a1.csv → human_s1.csv → robot_s1.csv

   human_a2.csv → human_s2.csv → robot_s2.csv

Hints to complete robot_execute_under_policy function

Open up the python code: `planner.py`

```
def robot_execute_under_policy(gridworld, robot_init_state, policy, human_states):
        robot_states = []
############## to be completed here; XXXX are parts to be edited ##############
        # print(f "policy {policy[(2, 0, 0, 1, 1)]}") #for python3.6
        # print("policy {}".format(policy[(2, 0, 0, 1, 1)]))

        rs = robot_init_state

        for hs in human_states:
                query = (XXXX, XXXX, XXXX, XXXX, XXXX) #include the robot position states, human position states, followed by human request status; refer to hrimdp.py for clues

                #print(f'query = {query}') #for python3.6
                print("query = {}".format(query))

                current_policy = policy[XXXX]           # update policy
                current_action = robot_actions[XXXX]    # update robot actions
                next_cell = [c + a for c, a in zip(XXXX, XXXX)] # update robot position states

                if gridworld.is_wall(next_cell[0], next_cell[1]):       # robot is hitting wall
                        robot_states.append((XXXX, XXXX))       #state remains the same
                else:                                           # robot is not hitting wall
                        rs = next_cell          # Update rs variable with new values
                        robot_states.append(rs) # Update robot position states with new values

#######################################################################
```

DEMO

playback

simulate system dynamics

visualization

Part B: Testing of Developed Algorithms on Various Virtual Worlds

Things to do

1. **Only for the world_room map (provided)**, handcraft **2 different human_actions_hand.csv** with meaningful interactions: *human_a_h1.csv* and *human_a_h2.csv*

    e.g. smart navigation using world_room.csv

python3 human_execute.py world_room.csv human_a_hX.csv 0 0 1 human_s_hX.csv

python3 planner.py world_room.csv human_s_hX.csv 10 9 robot_s_hX.csv

python3 visualizer.py world_room.csv human_s_hX.csv robot_s_hX.csv

Hint: Open (with gedit) to view the world_room.csv file. Assume that the human starts at [0,0] (i.e. the top left corner of the map), you can plan the actions by editing the *human_a_h0.csv* template given by "moving" the human (using the S, R, L, U, D keys) through the zeros (i.e. the empty spaces) in the map. You may include the T key (i.e. Toggle Request Key) at your preference in between the actions.

Things to do

2. Complete function **random_actions** in **random_human_actions.py**

3. For each of the 3 environments (provided; *world_XXX*), randomly generate **TWO human_actions_random.csv**, e.g.

python3 random_human_actions.py 10 100 human_a_rX.csv

python3 human_execute.py world_XXX.csv human_a_rX.csv 0 0 1 human_sX.csv

python3 planner.py world_XXX.csv human_sX.csv 10 9 robot_sX.csv

python3 visualizer.py world_XXX.csv human_sX.csv robot_sX.csv

*You have to assign **X** with the correct parameters*

4. Observe the interactions between the human and robot within the various worlds as well as based on the various human action sets

Hints to complete random_actions() function

Open up the python code: `random_human_actions.py`

random_human_actions.py (~/Desktop/HRSE/workshop/day3/day3b) - gedit

Open ▾ Save

```
def random_actions(seed, timestamps):
        actions = []

        random.seed(seed)
        np.random.seed(seed)

########## to be completed; replace XXXX with correct terms ##########

        # make sure human actions follow the probabilities set in the hrimdp model
        human_actions = [XXXX]
        human_prop = [XXXX]
        actions = np.random.choice(human_actions, timestamps, human_prop).tolist()

######################################################################

        return actions
```

DEMO

playback

simulate system dynamics

visualization

Instructions

This is a group project. Each group representative submits one zip file of all your codes/files (i.e. py, CSV) into Canvas at the end of the workshop

HRSE_Project3.zip

- Download all files in the directory **/workshops/day3** for reference codes
- Refer to the README file for instructions

Optional
(If you completed early)

1. First, create 2 large (at least 20x20) gridworld environments (i.e. world_custX.csv)
2. For each of the 2 environments (created; *world_XXX*),
   a. handcraft **2 human_actions_hand.csv** with meaningful interactions:
      e.g. smart navigation through the map
   b. randomly generate **2 human_actions_random.csv** with different initial human/robot positions, e.g.

python3 random_human_actions.py 10 100 human_a_rX.csv

python3 human_execute.py world_XXX.csv human_a_rX.csv 0 0 1 human_sX.csv

python3 planner.py world_XXX.csv human_sX.csv 10 9 robot_sX.csv

python3 visualizer.py world_XXX.csv human_sX.csv robot_sX.csv

3. Observe the interactions between the human and robot within the various worlds as well as based on the various human action sets

THANK YOU

Email: nicholas.ho@nus.edu.sg