Simple & Compact

Group F

Team Mammonite

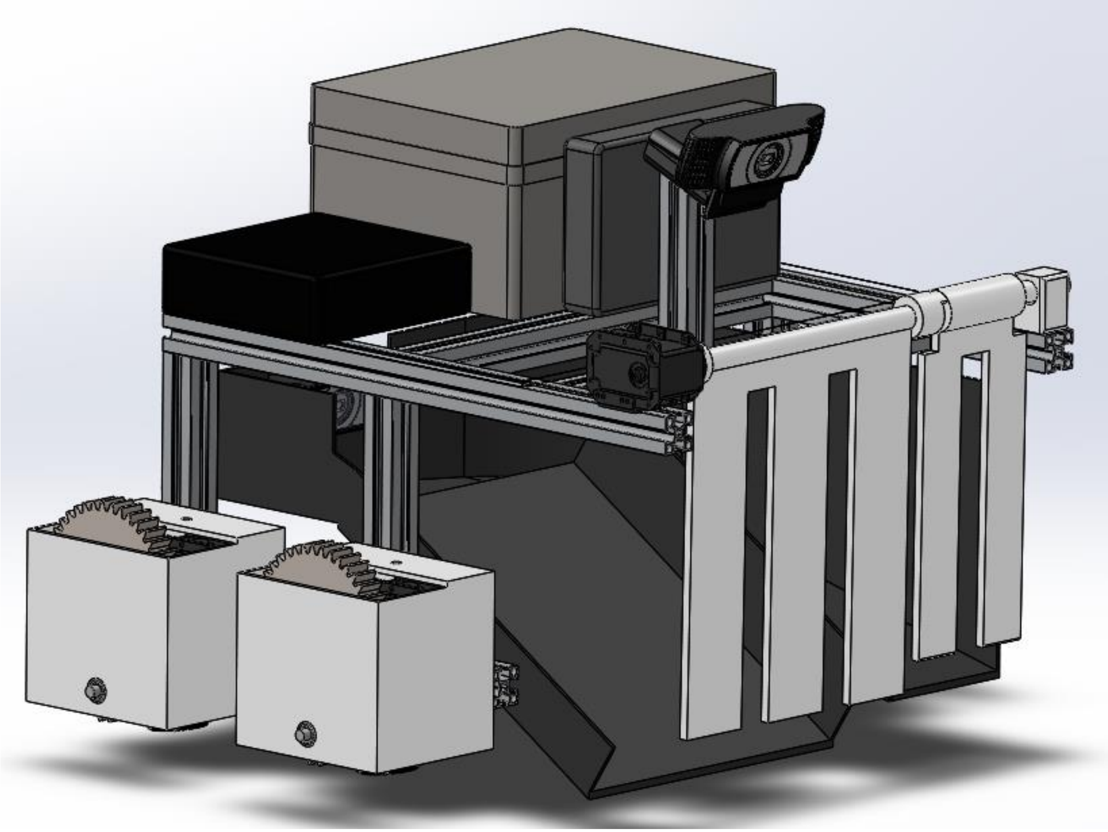
맘모나이트

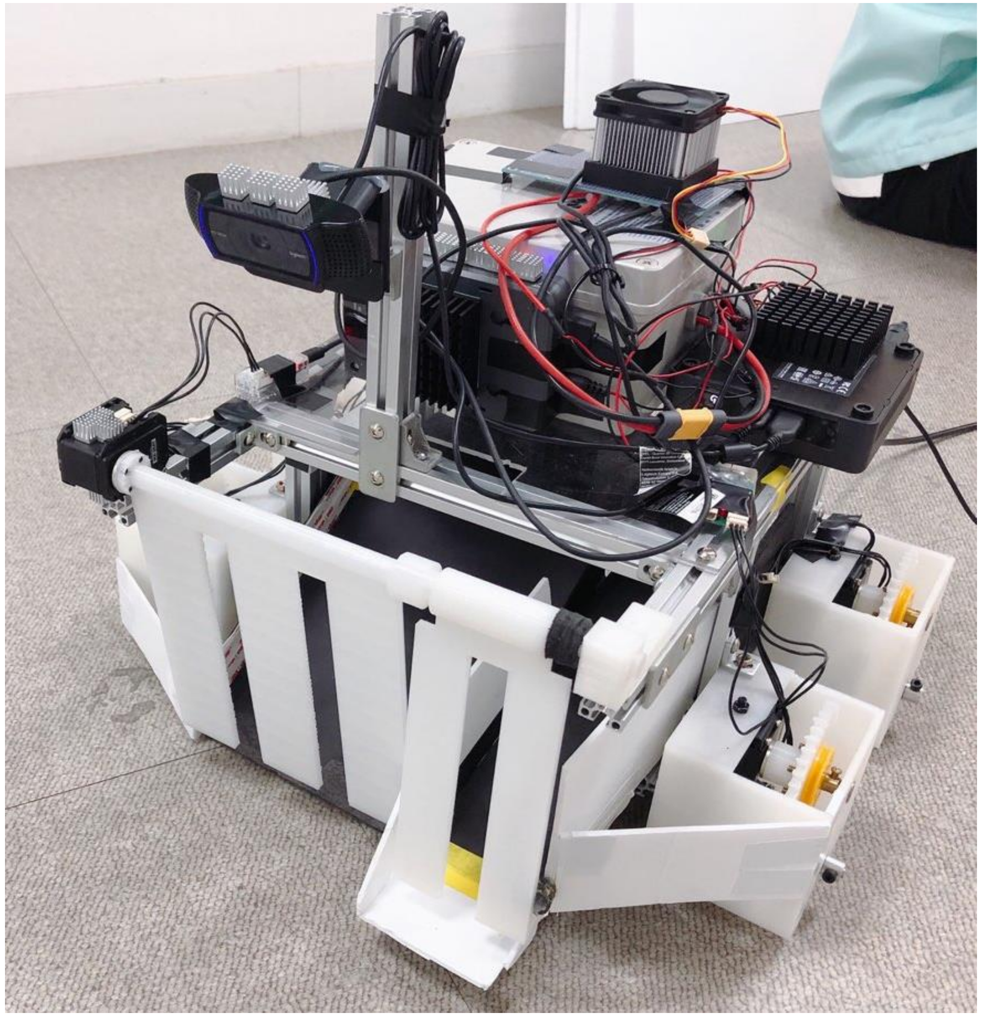
Professor : Park Youngjin

Presentor : Lee Jongeun

Member : Jung Mingi Lee Inpyo Lee Geonwoo Lee Jongeun Yeom Donghoon Hwang Seokyoung

Overview





System Design

Entire Design

Thermal Design

Motor Control

Control Process

Vision Processing

Calibration

Performance

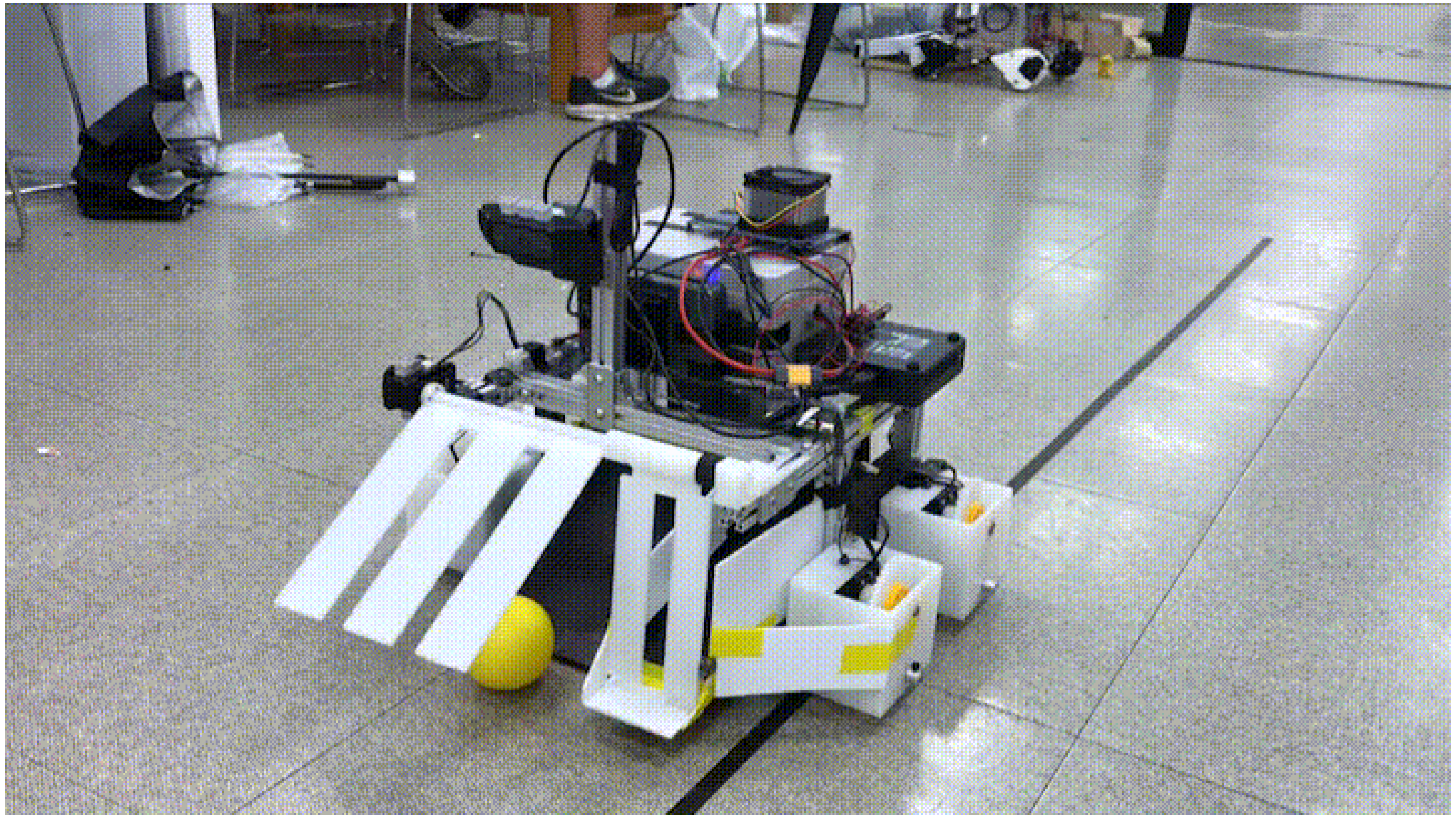
System Integration

ROS Structure

Algorithm

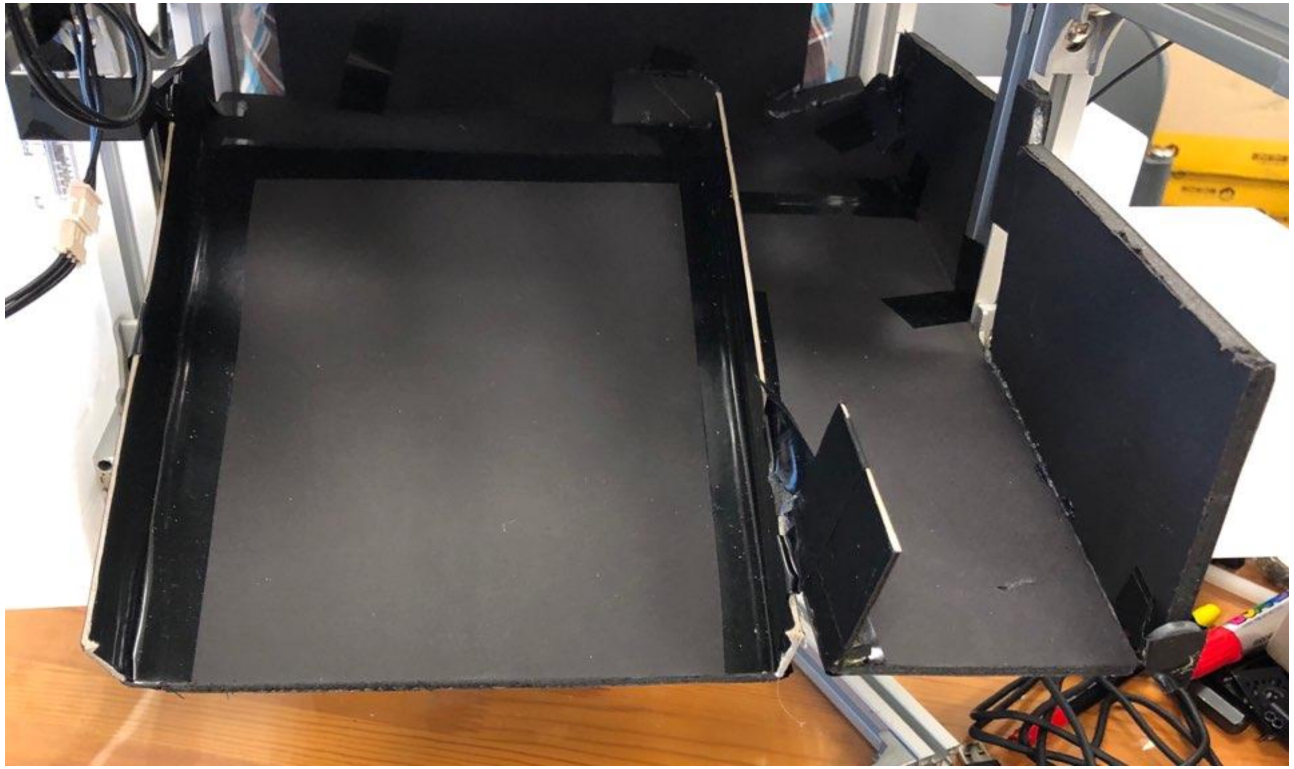
System Design

System Design - Picking & Releasing



1. Use only 1 motor
2. No direction change for parking

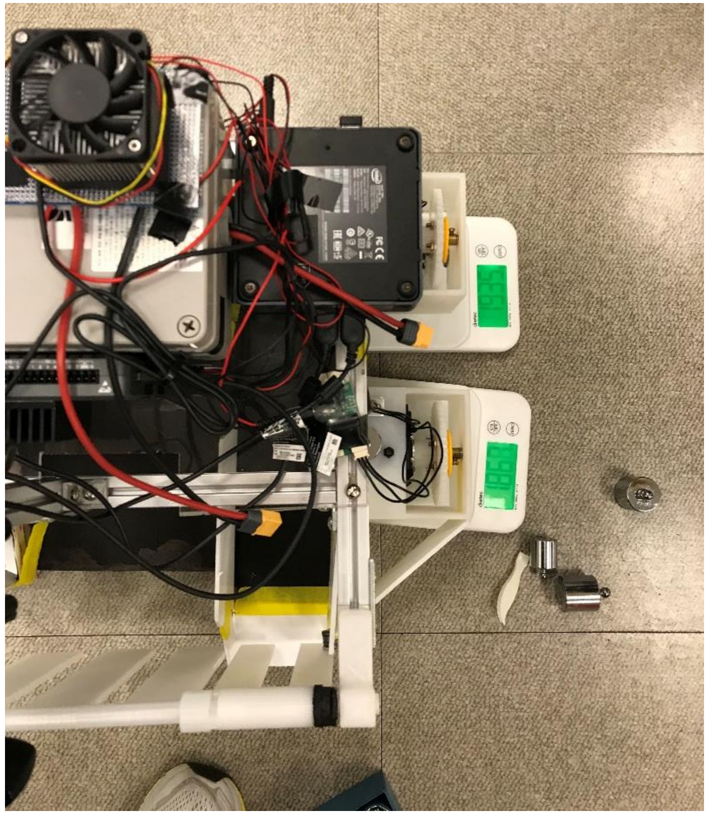
System Design - Material



<Cardboard Paper>

88g, 0.09697g/cm^3

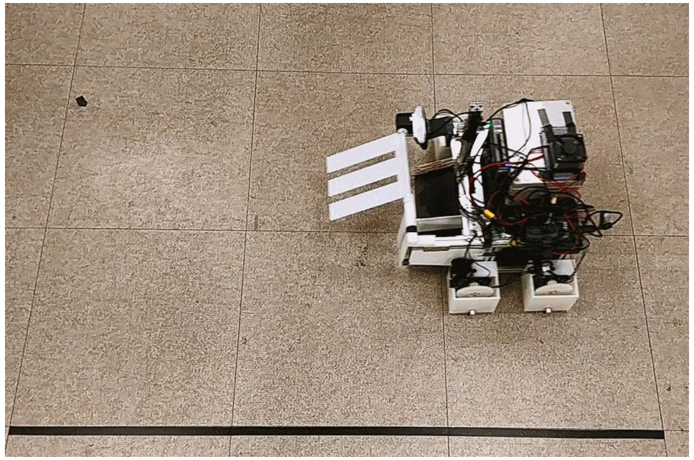
System Design – Weight adjustment



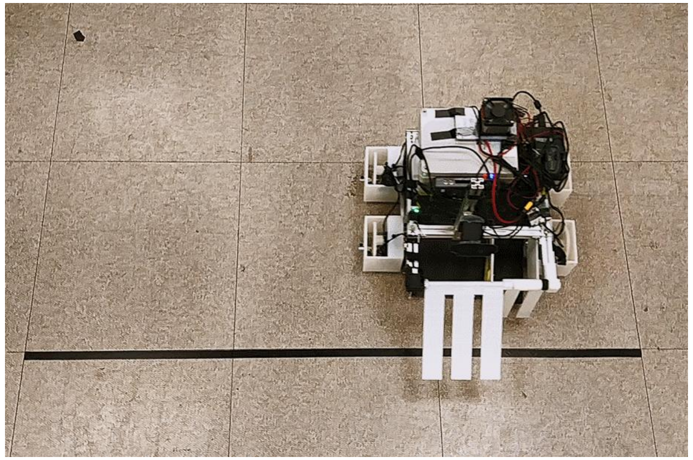
2047g	1935g
1922g	1867g

Average	1942.75g
Total	7771g
Difference	-3.8%~5.3%

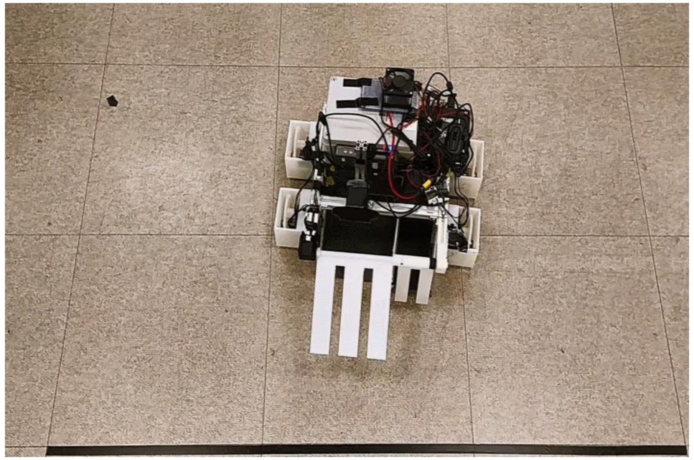
System Design – Moving



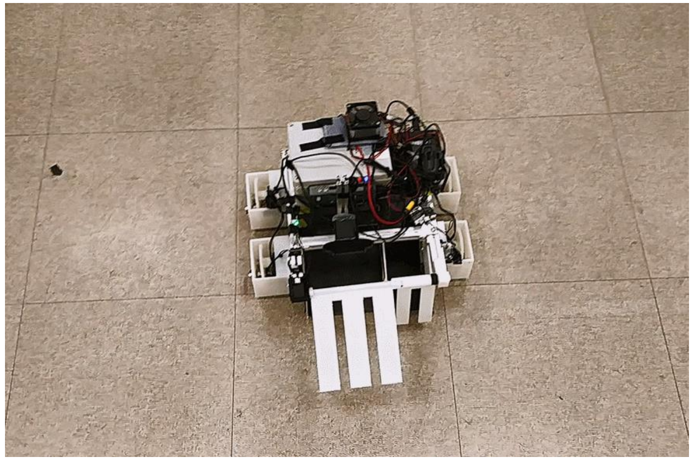
<Straight>



<Left and Right>



<CW rotating>



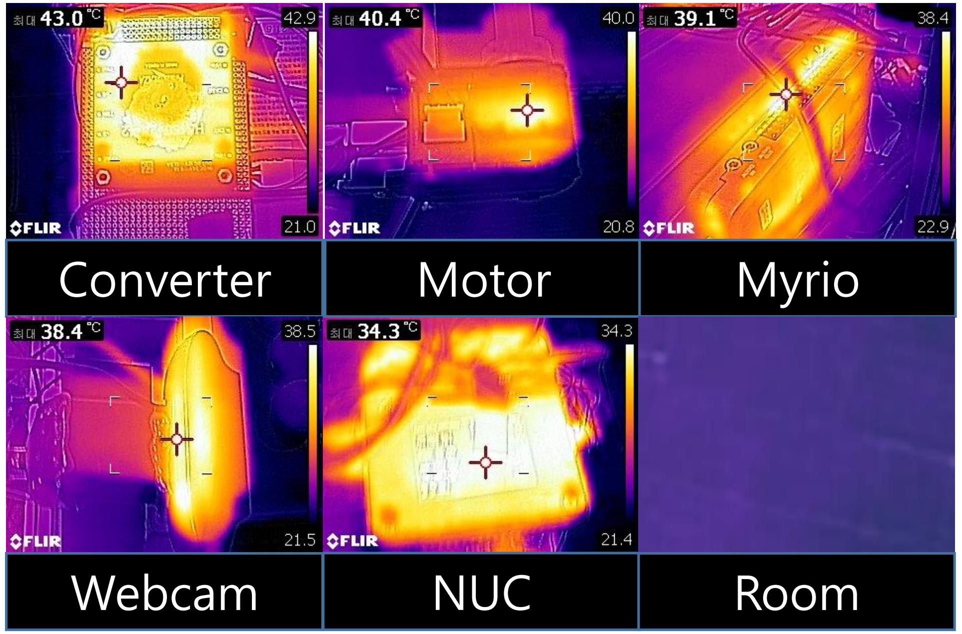
<CCW rotating>

System Design – Heat Distribution

5min operation without heat distribution

Room Temperature = 21.4℃

Part	Temperature(℃)
Converter	**43.0**
Pickup Motor	**40.4**
Myrio	**39.1**
Webcam	**38.4**
NUC	**34.3**



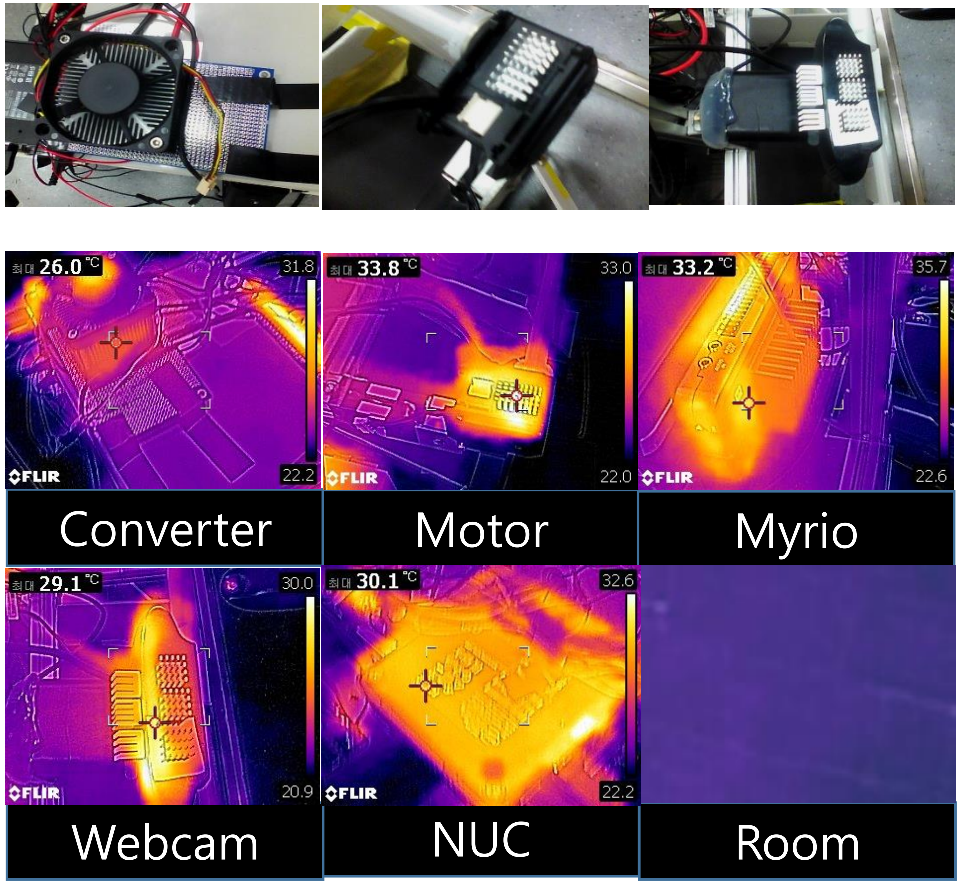
~~Duct~~ → Fan & Fin

System Design – Heat Distribution

5min operation with fan & fin

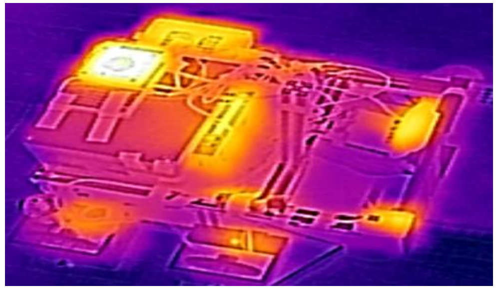
Room Temperature = 21.4℃

Part	Temperature(℃)
Converter (Fan&Fin)	**26.0**
Pickup Motor (Fin)	**33.8**
Myrio (Fin)	**33.2**
Webcam (Fin)	**29.1**
NUC (Fin)	**30.1**



System Design – Heat Distribution

Before



After

Part	Temperature difference with room temperature(ΔT, ℃)	
	Without Fan&Fin	With Fan&Fin
Converter (Fan&Fin)	21.6	4.6
Pickup Motor (Fin)	19.0	12.4
Myrio (Fin)	17.7	11.8
Webcam (Fin)	17.0	7.7
NUC (Fin)	12.9	8.7

Motor Control

Motor Control - Control Process

ROS

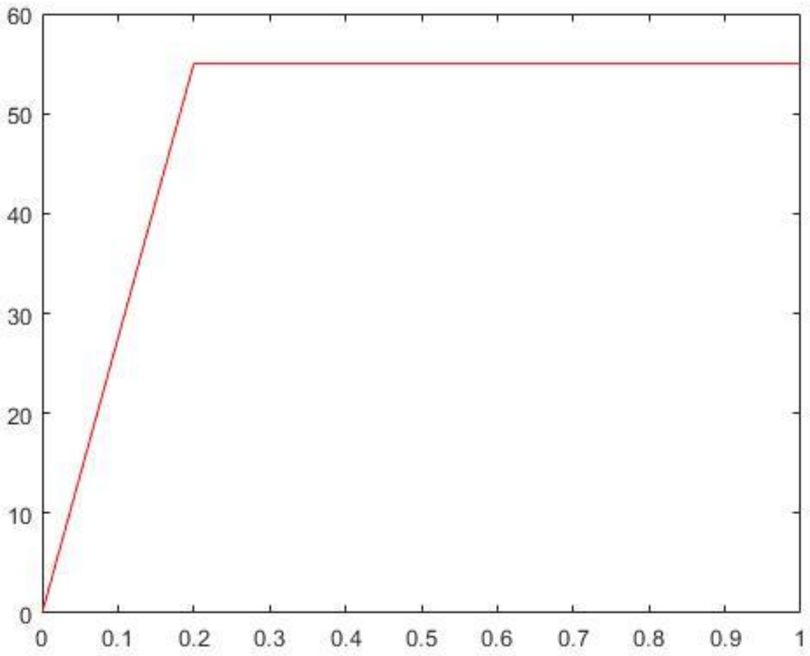
TCP/IP

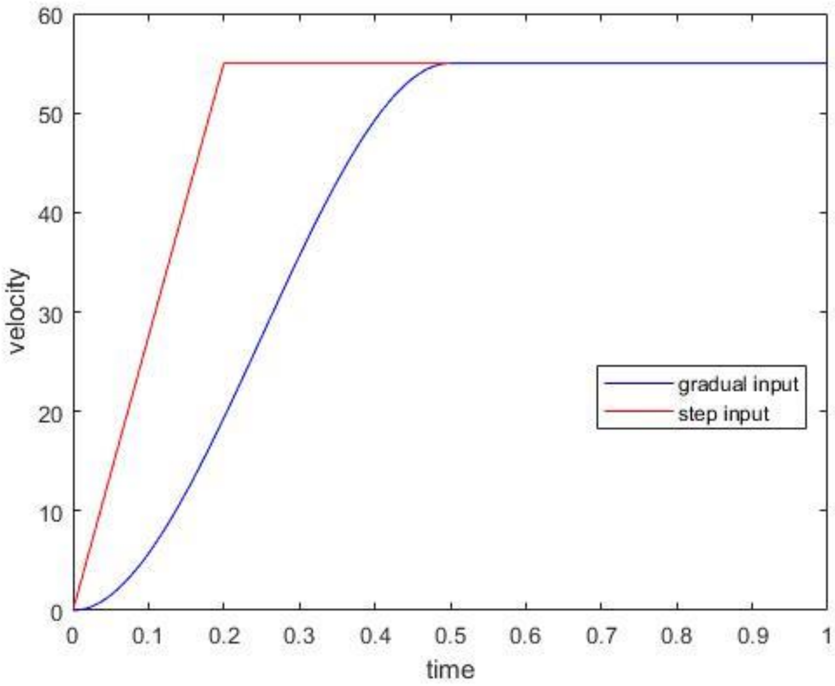
Phase

Desired RPM

Phase = 1 Move forward Phase = 2, 3 Move left & right Phase = 4, 5 Turn	Phase = 6 Pick up Phase = 7 Park & Release

Motor Control – velocity input





Step input

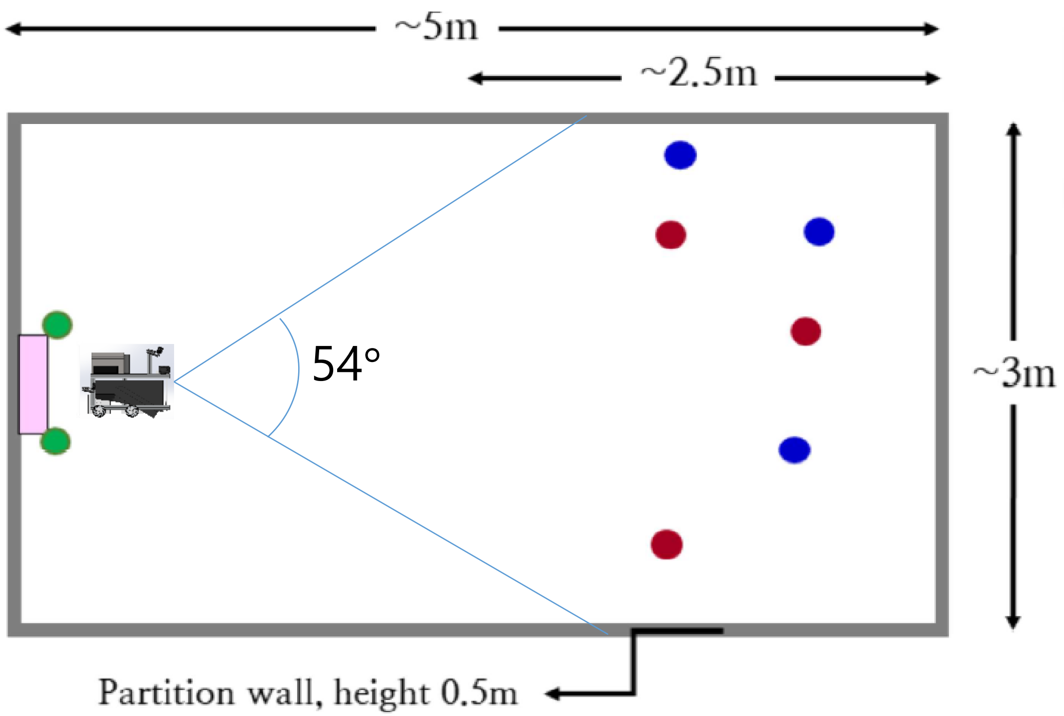
- Sudden acceleration change can induce motor damage
- It can cause vibration

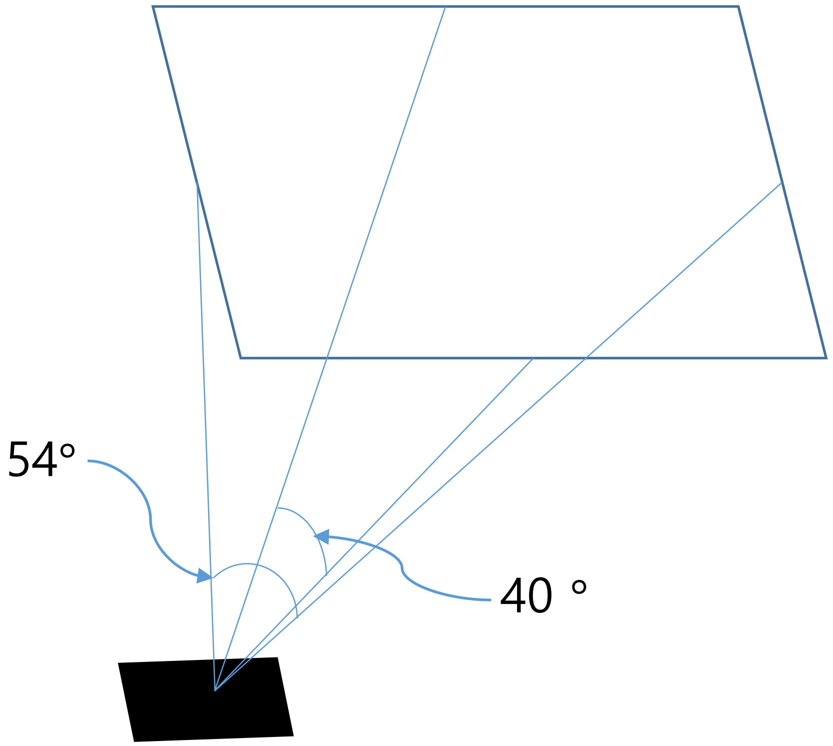
Gradual input

- Prevent sudden velocity change
- Spend more time to accelerate, but reduce error and precise

Vision Processing

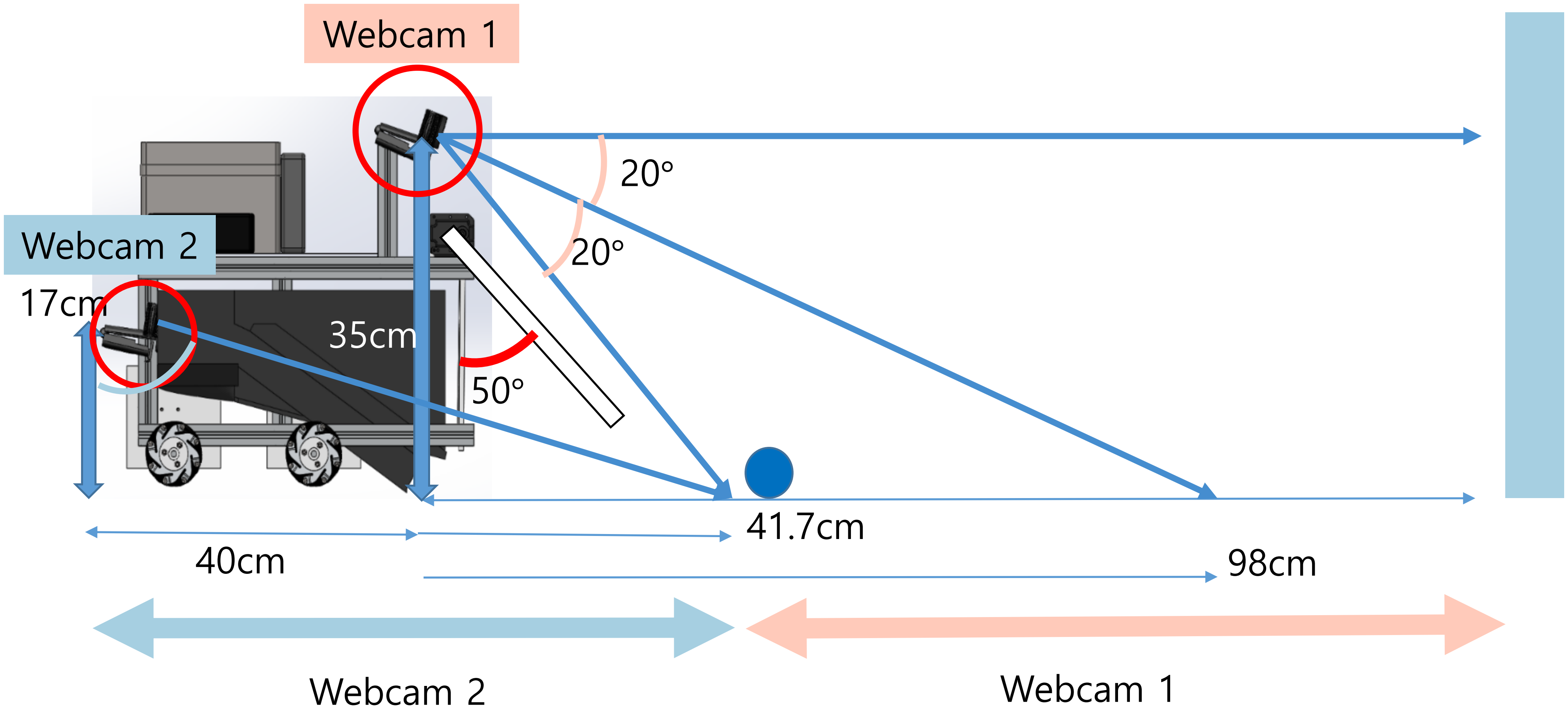
Vision Processing – Viewing angle of Webcam



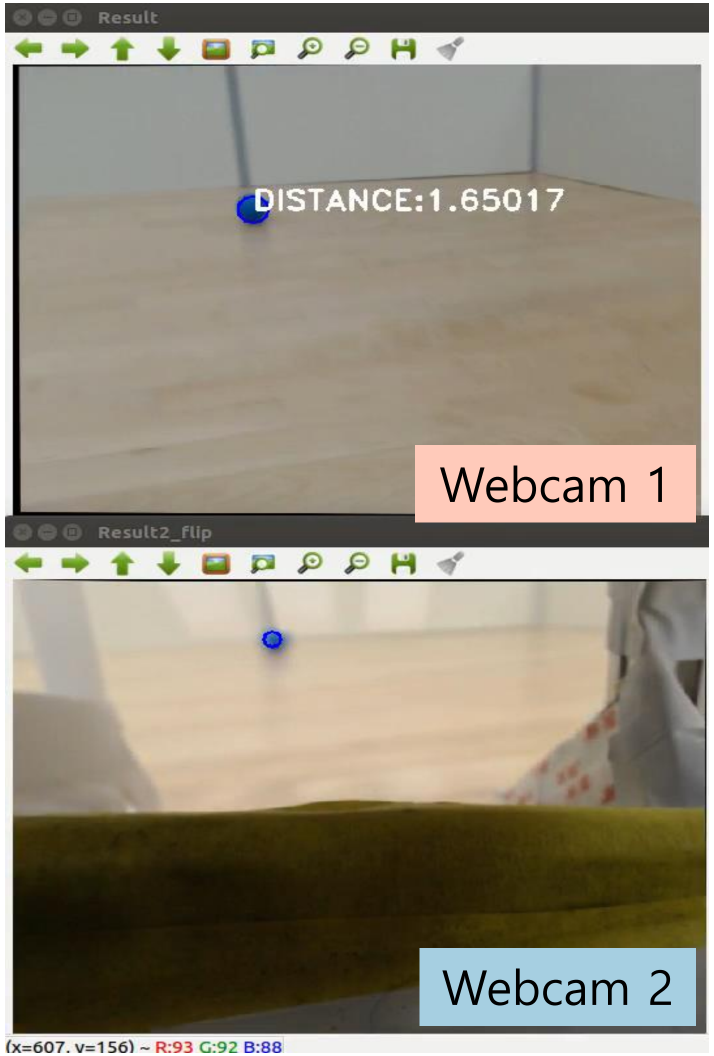


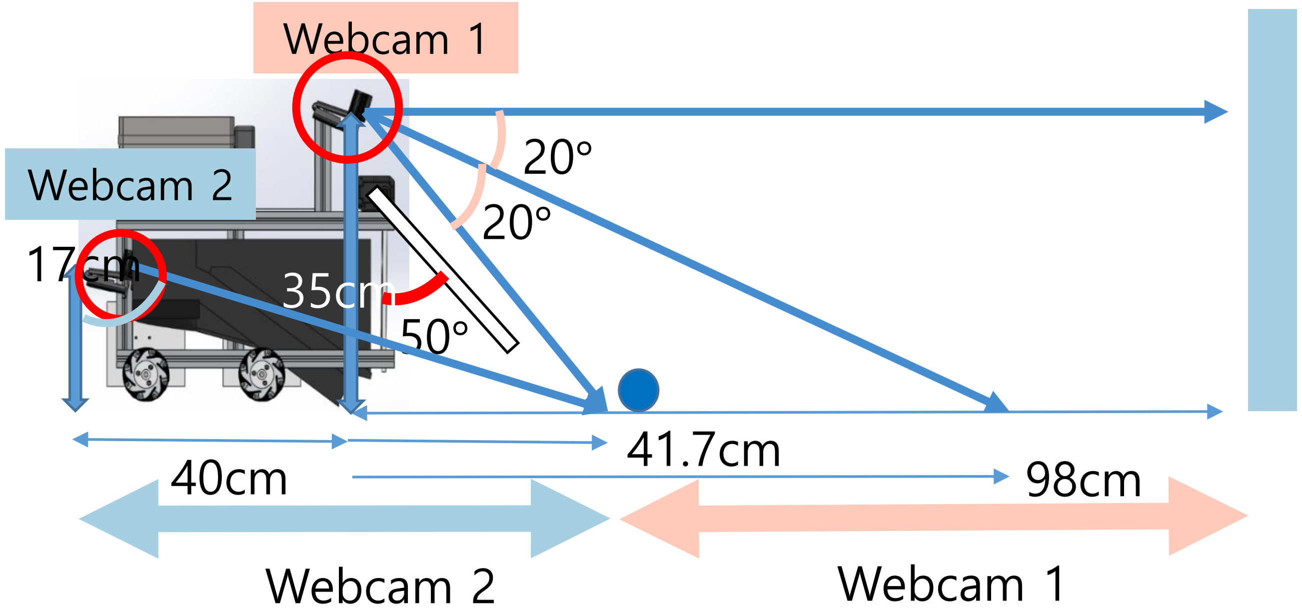
All balls can be detected in the initial position.

Vision Processing – Location of Webcam

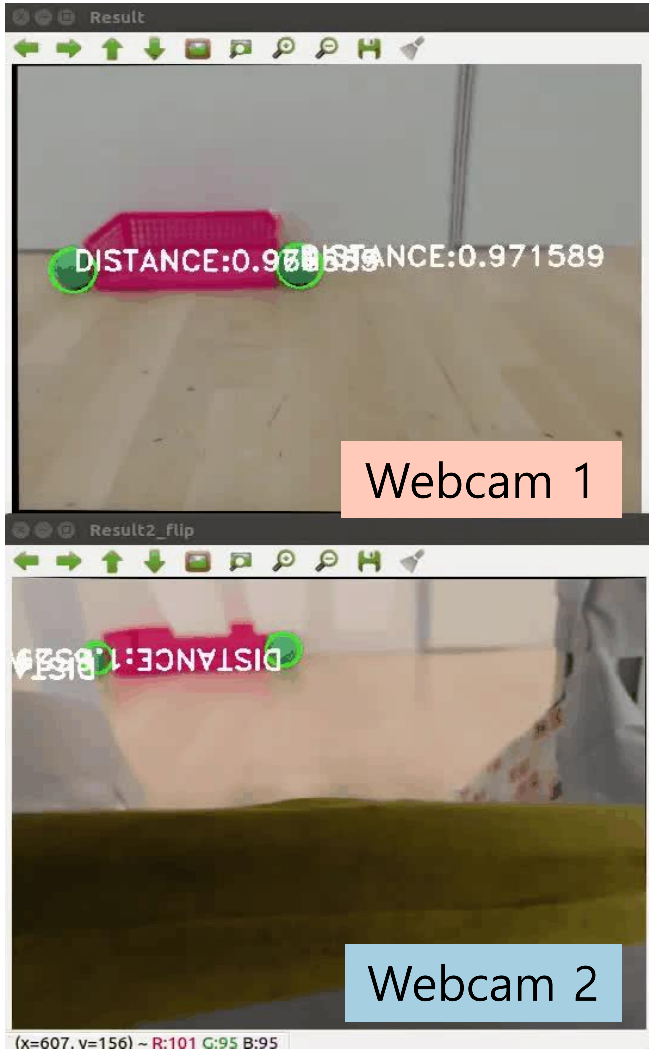


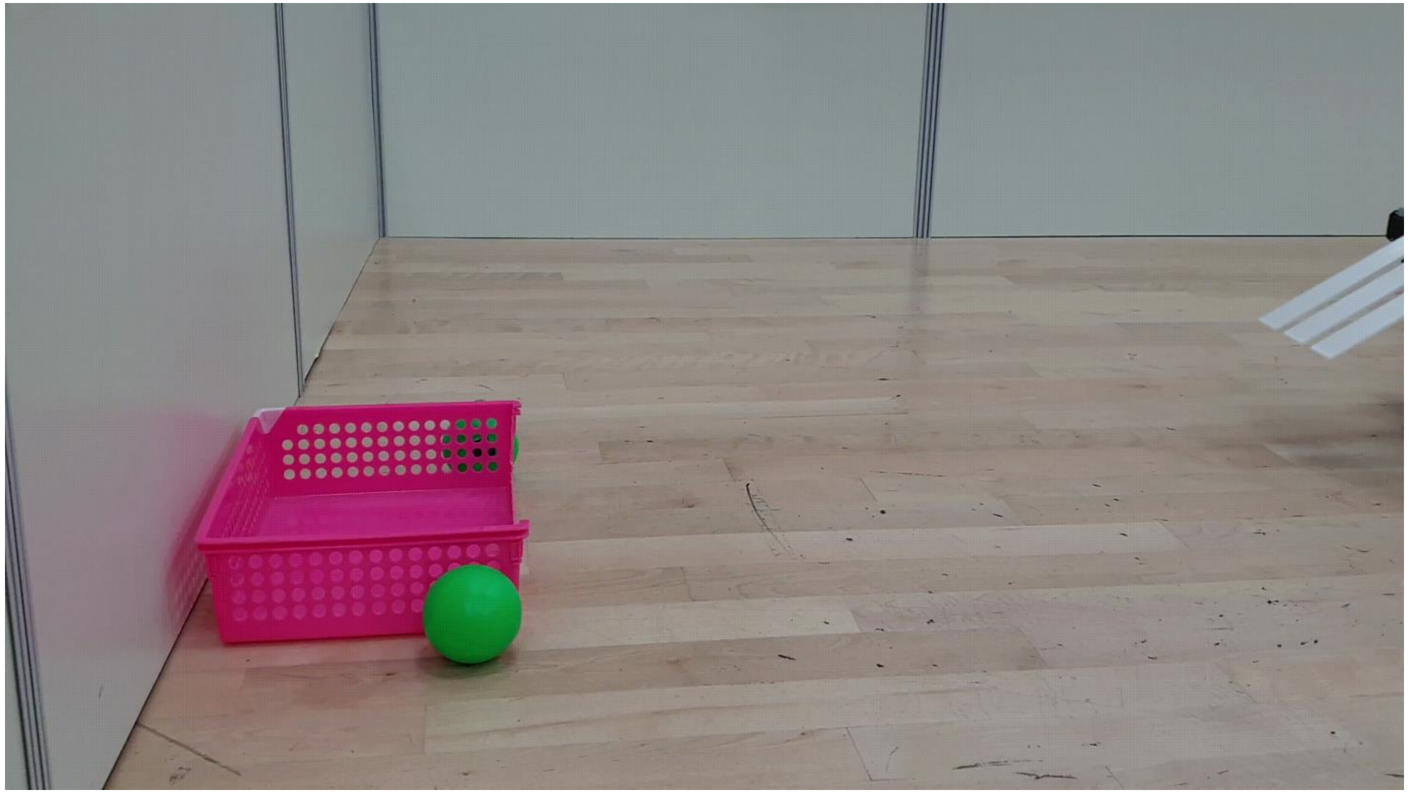
Vision Processing – Location of Webcam





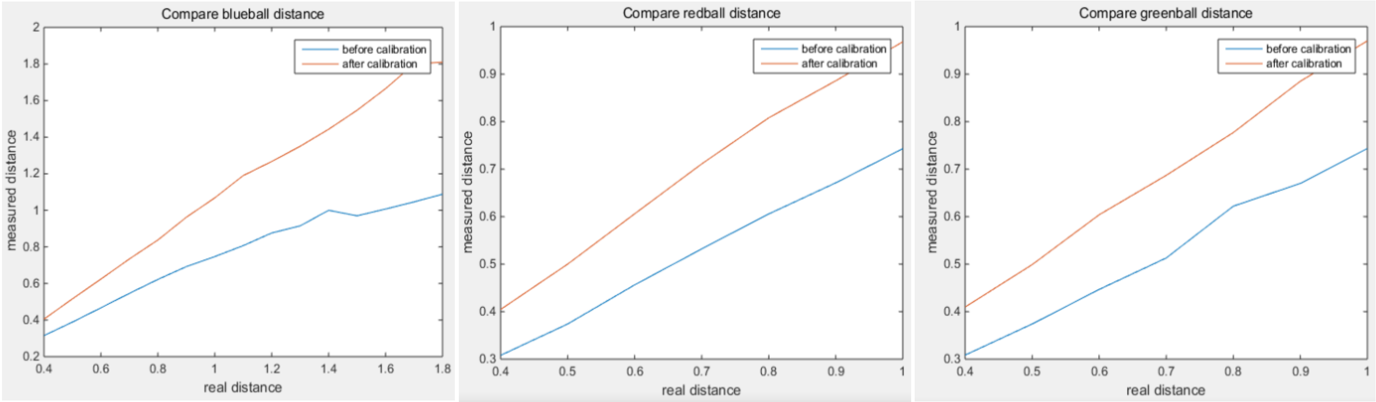
Vision Processing – Location of Webcam



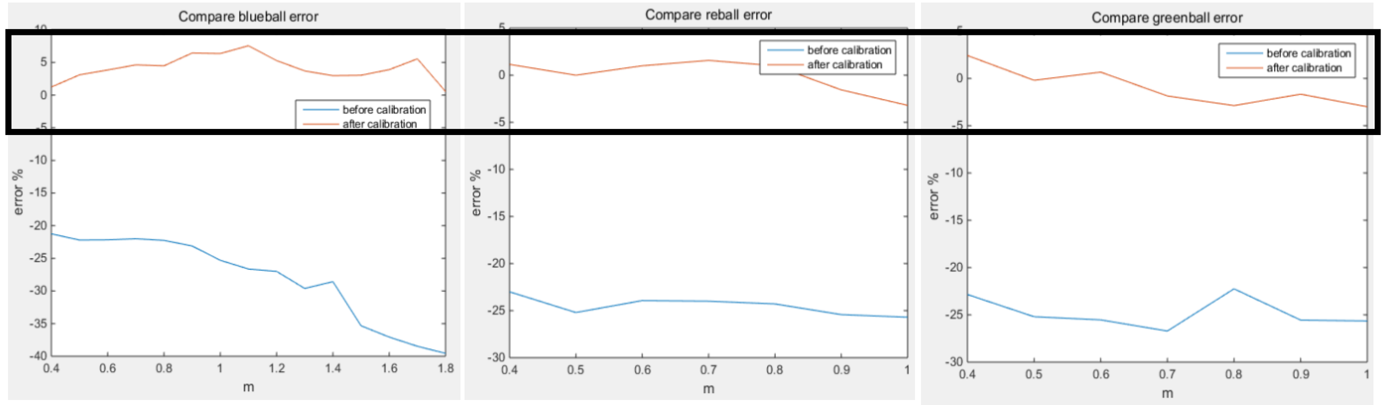


Vision Processing – Reducing error code

<Real Distance and Measured Distance>



<Error between Real Distance and Measured Distance>



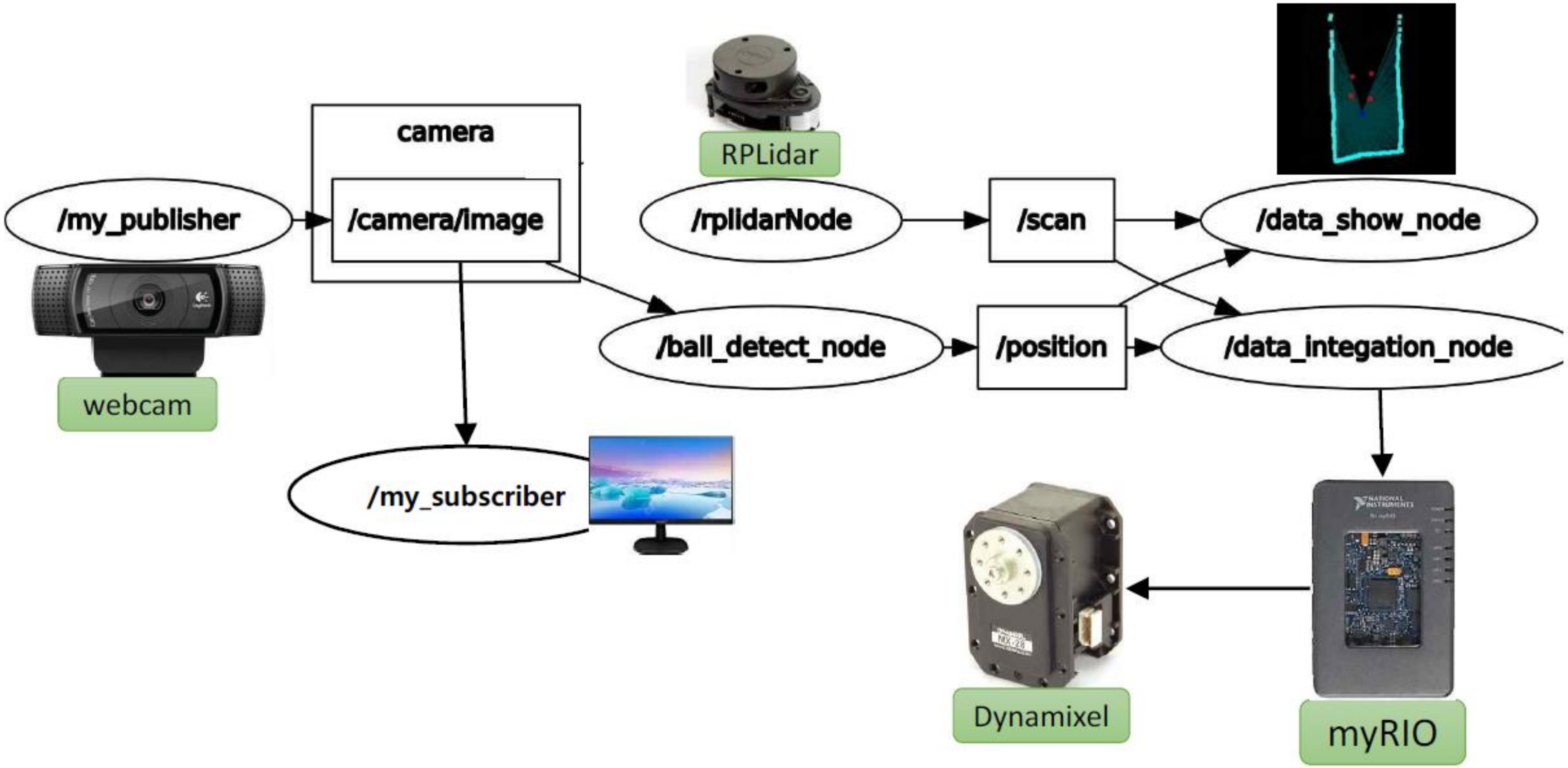
=> Make error less than 5% every color of ball

Ball seems larger than reality.
Use matlab polyfit for fitting.

```
for(size_t i = 0; i <contours_r.size(); i++){
approxPolyDP(contours_r[i], contours_r_poly[i], 3, true);
minEnclosingCircle(contours_r_poly[i], center_r[i], radius_r[i]);
radius_r[i] = 0.7536*radius_r[i]+0.6771;
}

for( size_t i = 0; i < contours_b.size(); i++ ){
    approxPolyDP( contours_b[i], contours_b_poly[i], 3, true );
    minEnclosingCircle( contours_b_poly[i], center_b[i], radius_b[i] );
    radius_b[i] = 0.8538*radius_b[i]-4.0814;
}

for( size_t i = 0; i < contours_b2.size(); i++ ){
    approxPolyDP( contours_b2[i], contours_b_poly2[i], 3, true );
    minEnclosingCircle( contours_b_poly2[i], center_b2[i], radius_b2[i] );
    radius_b2[i] = 0.8538*radius_b2[i]-4.0814;
}

for(size_t i = 0; i < contours_g.size(); i++){
  approxPolyDP(contours_g[i], contours_g_poly[i], 3, true);
  minEnclosingCircle(contours_g_poly[i], center_g[i], radius_g[i]);
  radius_g[i] = 0.7224*radius_g[i]+1.8049;
}
```

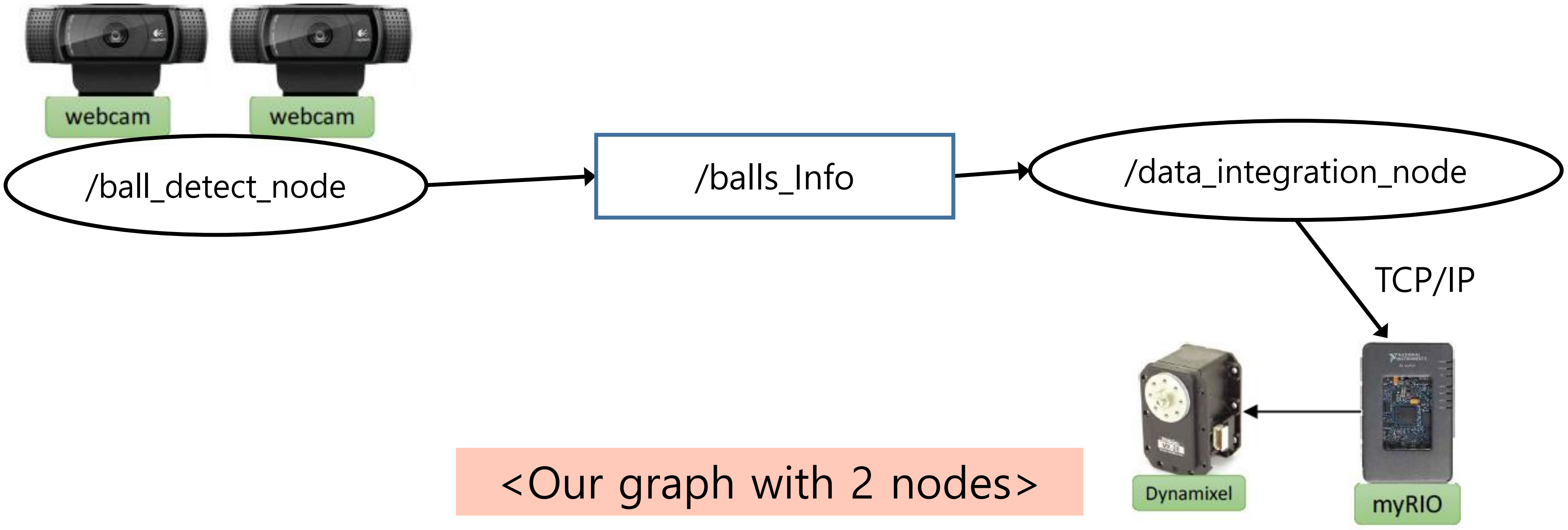
System Integration

System Integration - ROS Structure Diagram



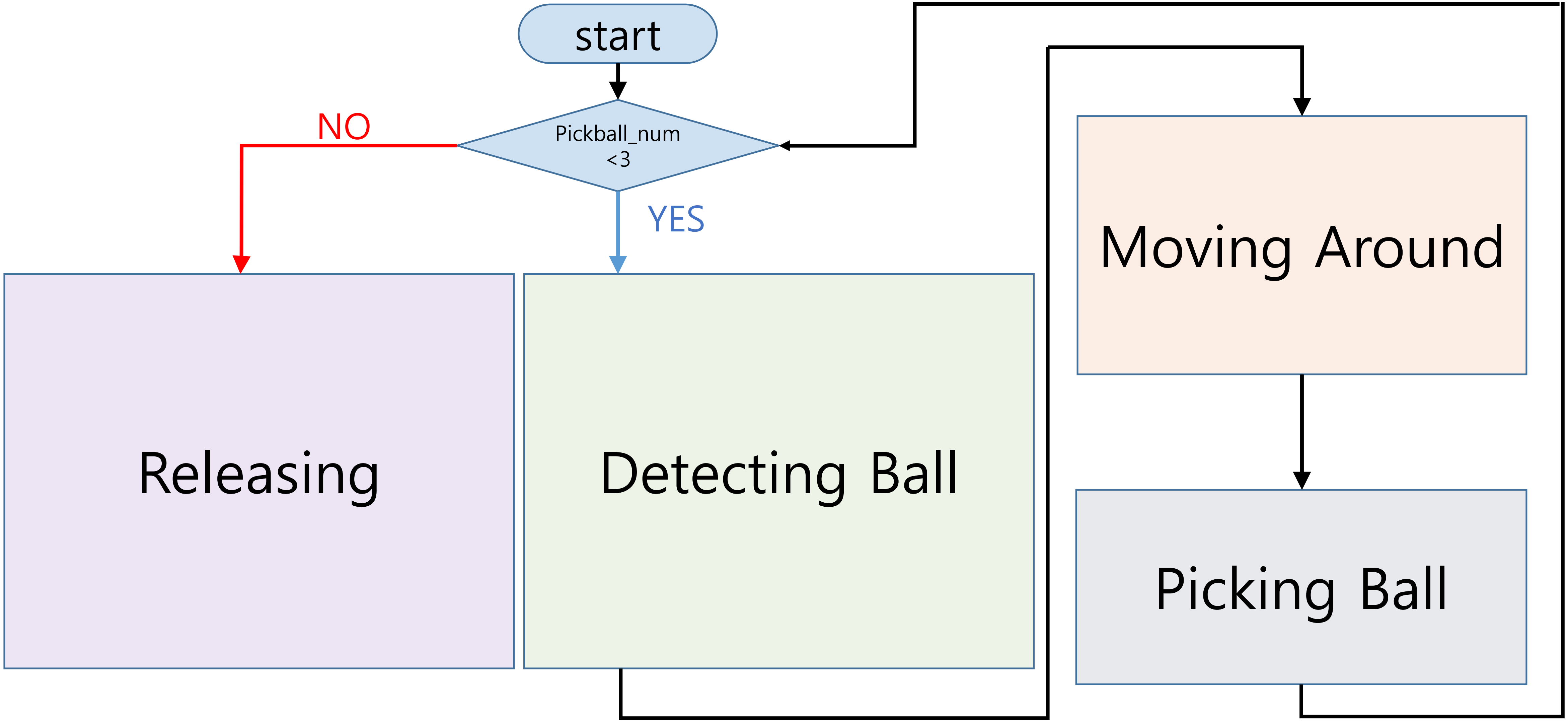
<Usual rqt_graph using Lidar with 6 nodes>

System Integration - ROS Structure Diagram

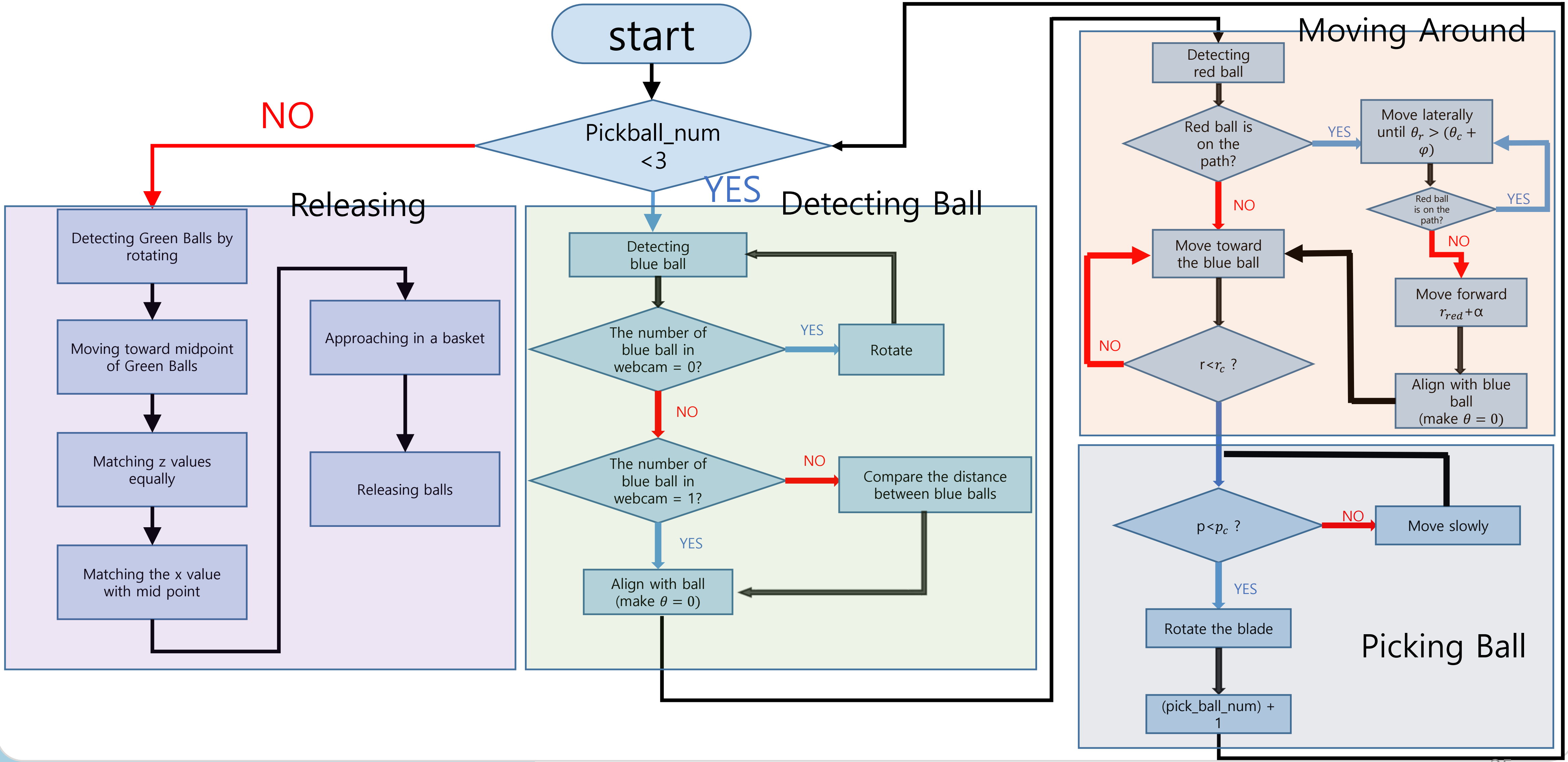


<Our graph with 2 nodes>

System Integration - Operation Logic Flows

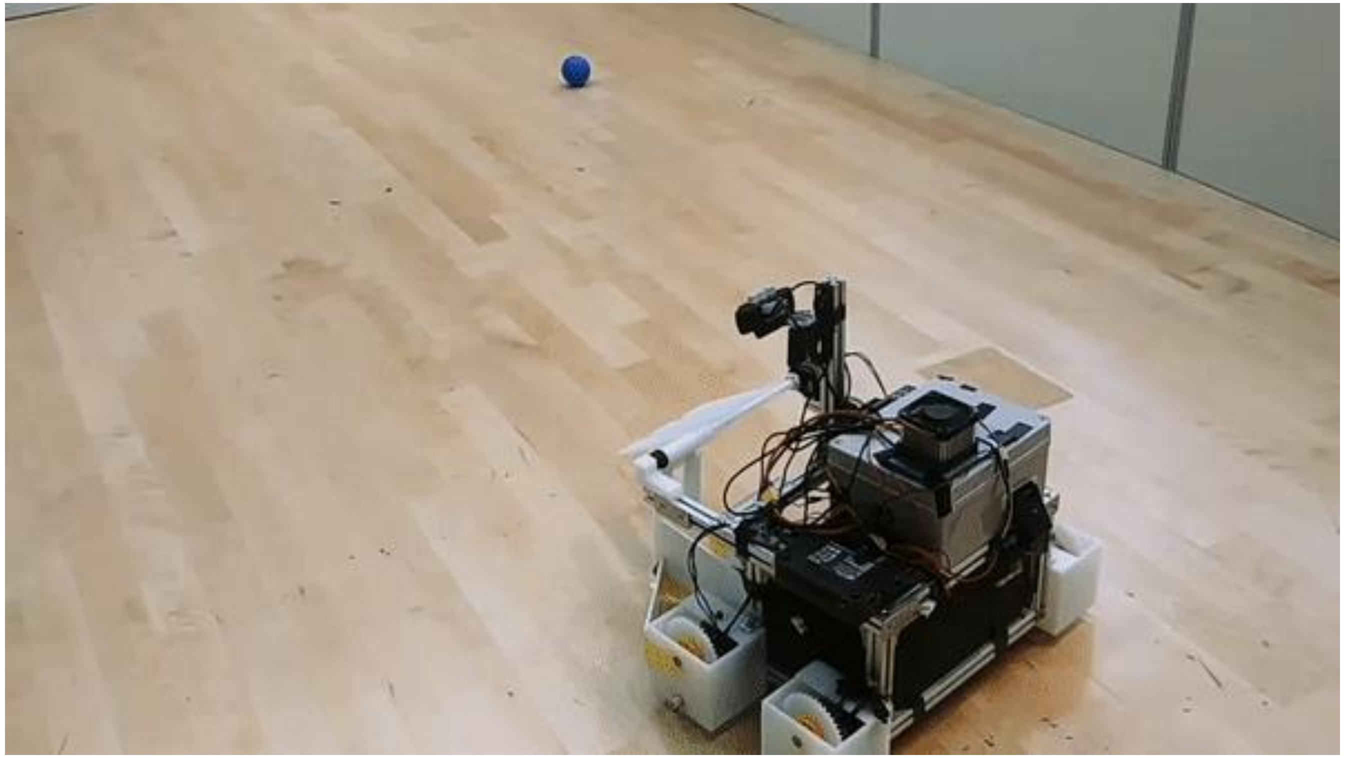


System Integration - Operation Logic Flows



System Integration - Operation Logic Flows

Detecting Ball

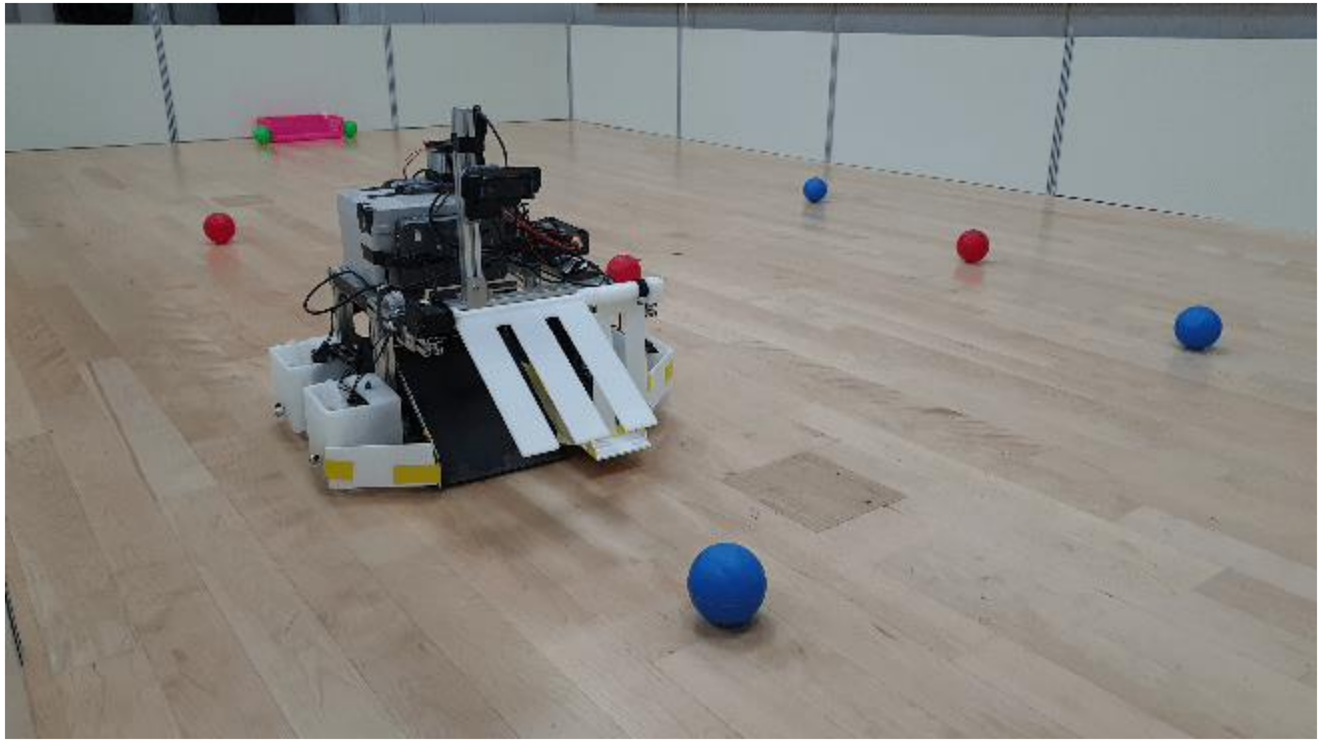


System Integration - Operation Logic Flows

Moving Around

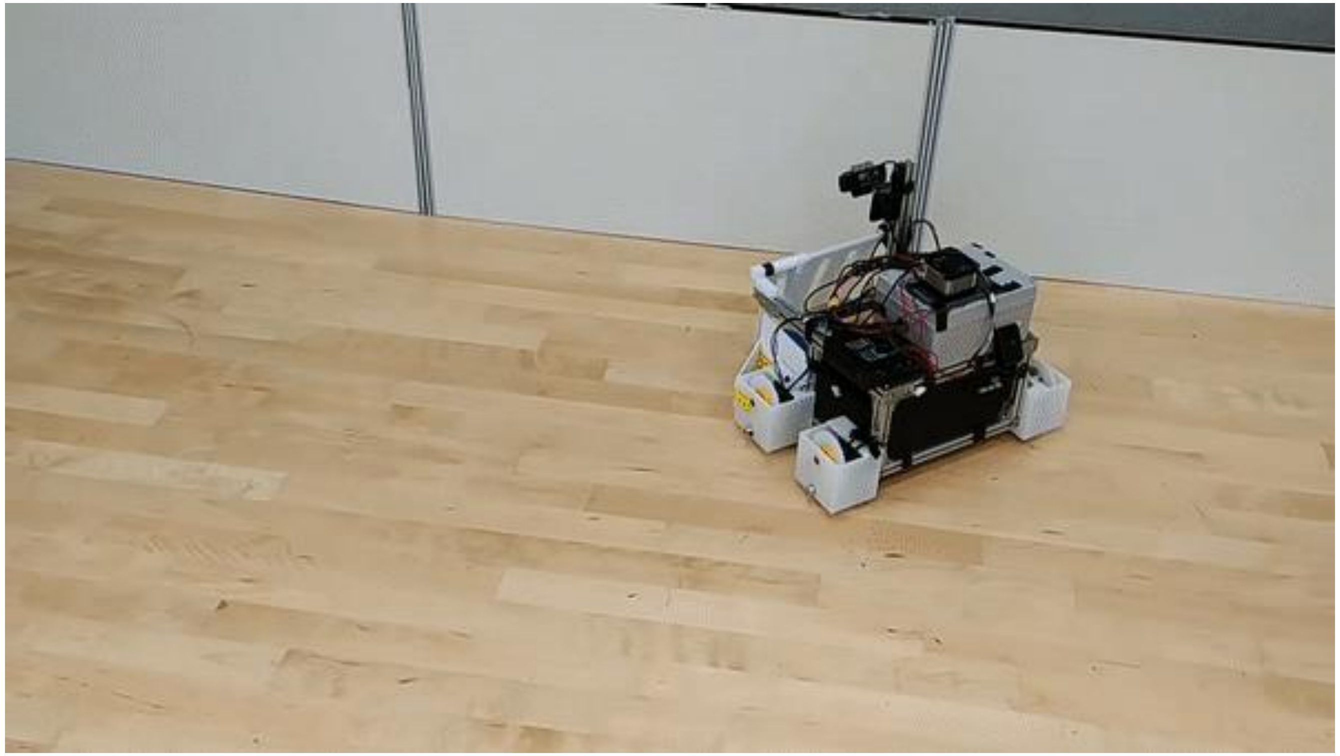
System Integration - Operation Logic Flows

Picking Ball



System Integration - Operation Logic Flows

Releasing



System Integration - Operation Logic Flows

Important feature of our algorithm : switch

```
while (ros::ok()){
  ros::spinOnce();
  If (pickup_ball_number < 3){
    switch (phase){
      case 1: Detecting Ball
      case 2: Moving Around
      case 3: Picking Ball
    }
  else {
    switch (phase){
      case 1: Detecting Green Ball
      case 2: Moving to Green Ball
      case 3: Releasing
    }
  }
```

System Integration - Operation Logic Flows

Important feature of our algorithm : switch

```
while (ros::ok()){
    ros::spinOnce();
    If (pickup_ball_number < 3){
        switch (phase){
          case 1:  Detecting Ball



    }
                                  }
```

System Integration - Operation Logic Flows

Important feature of our algorithm : switch

```
while (ros::ok()){
    ros::spinOnce();
    If (pickup_ball_number < 3){
        switch (phase){



            case 2:  Moving Around



    }
                                        }
```

Simple & Compact

Compact Hardware

7.7kg < 8kg

465*375*310(mm)

(Smaller and Lighter than last year models)

System Design

No suspension

Simple heat solution (than duct)

Pickup & Release

One Actuator

(Simple mechanism)

Frontside pickup & Release

Vision ROS

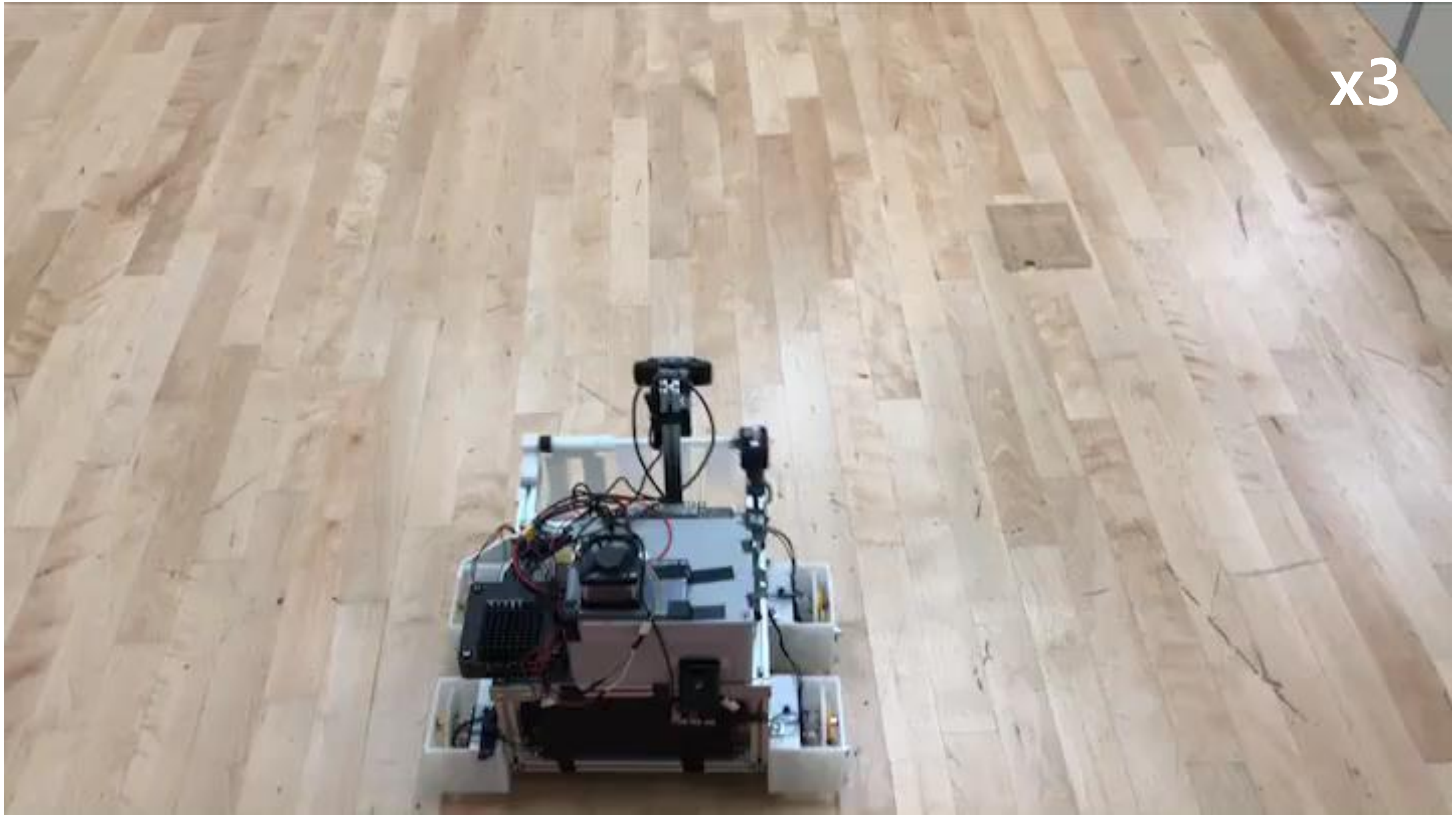
2 Webcams, but less sensor

2 Nodes < 6

Switch Statement

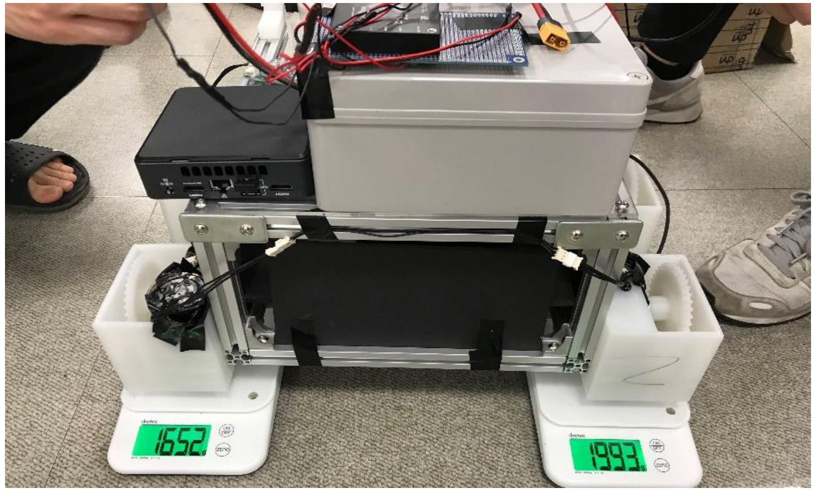
Demo Video

Pick up & Drop Off



Thank you

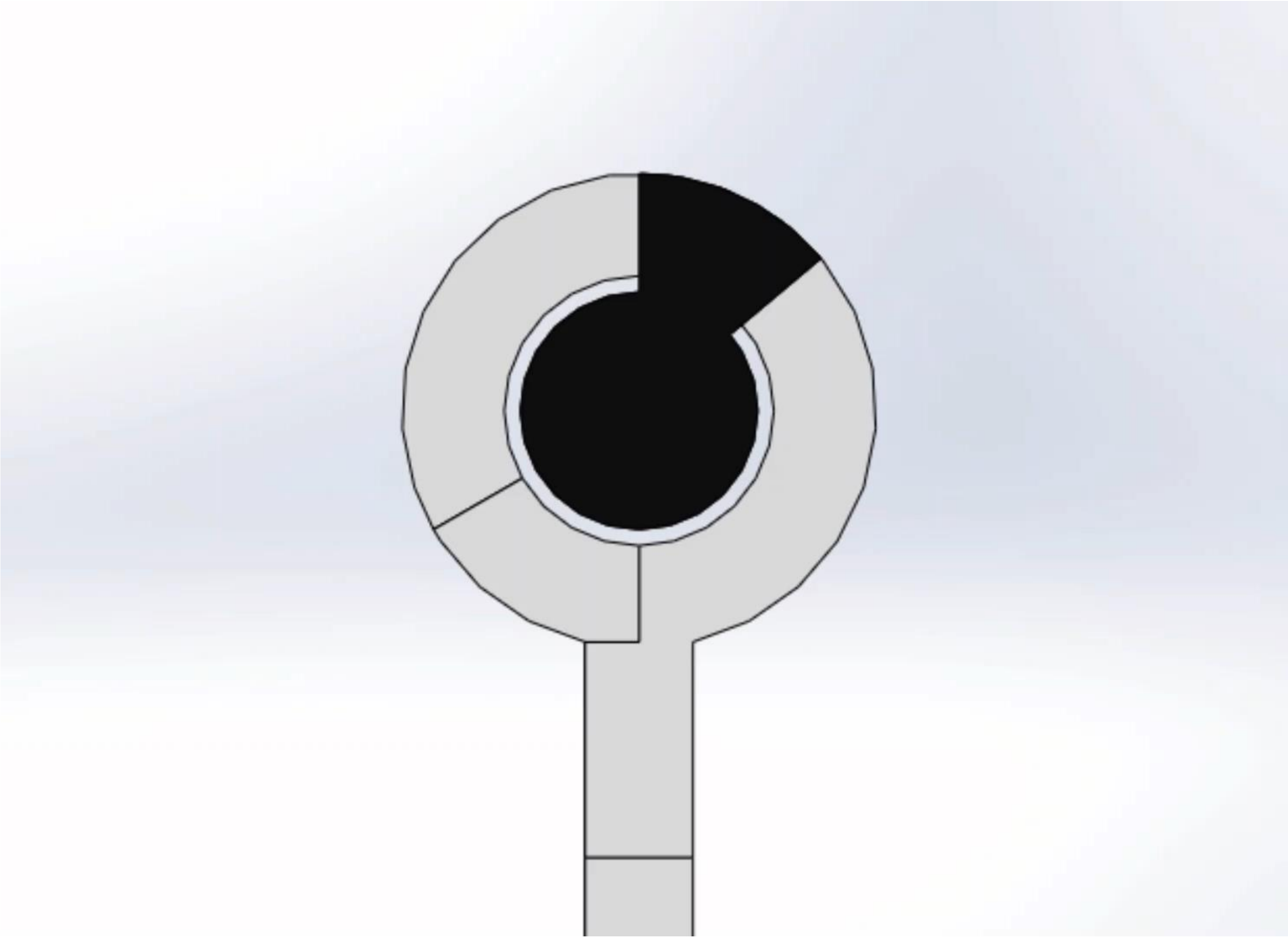
Normal Forces



1652g	1993g
1727g	1930g

Average	1825.5g
Total	7302g
Difference	5.3%~9.5%

System Design – Simple Mechanism



Motor Control - Gear Ratio Selection

1. Limitation of stall torque

$\tau_{mot} < \tau_{stall}$

$\Leftrightarrow$ N<5.102

2. Motor temperature should be lower than 35℃

(Time= 2min, safety factor=4.0, efficiency=0.1 (worst case))

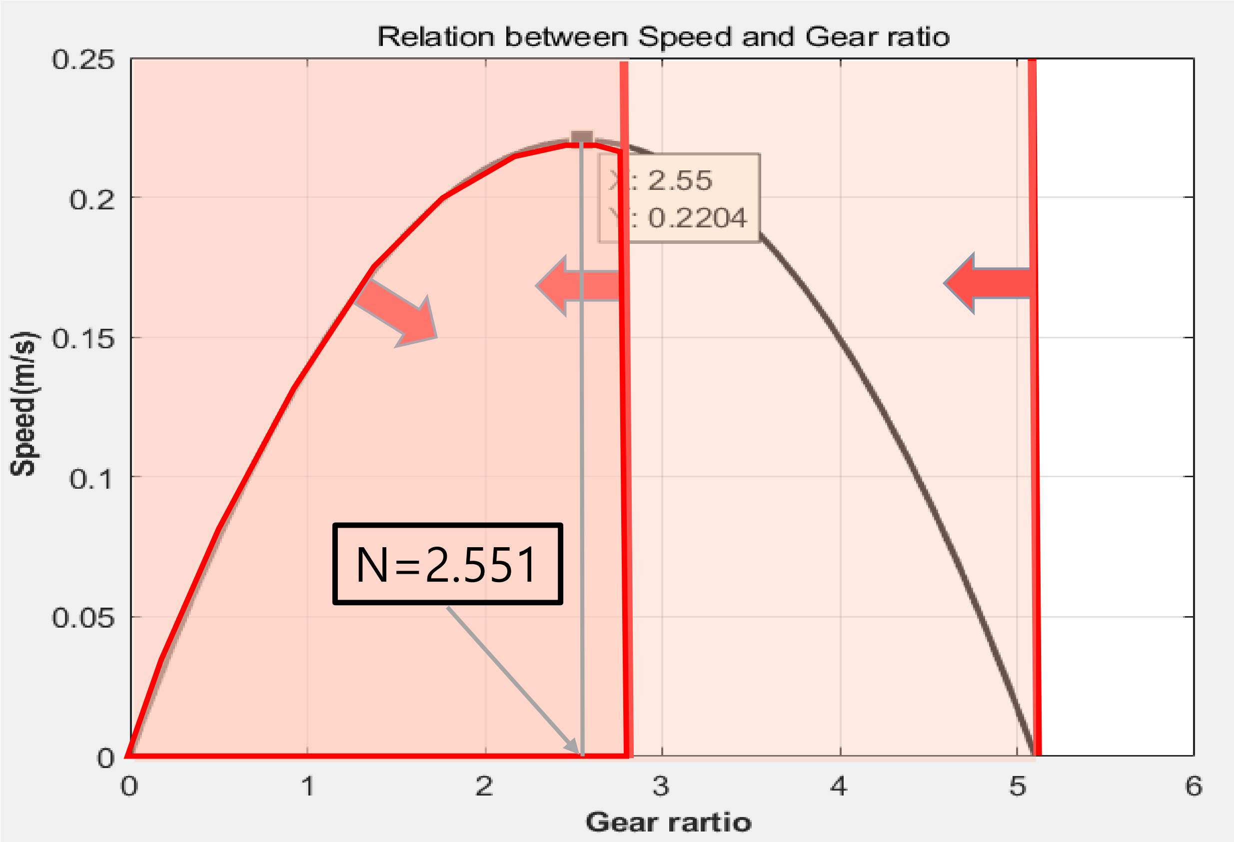
N<2.829

3. Maximize velocity

Motor speed $\propto N$ (at same angular velocity)

Angular velocity $\propto -N$ $(linearly\ decrease)$

$$V_{speed} = 0.17278 \cdot \mathrm{N} - 0.00338659 \cdot N^2$$



our gear ratio

15 : 40 =2.667

1. Motor torque is smaller than stall torque

$\tau_{mot} < \tau_{stall}$

$\Leftrightarrow \tau_{friction} \cdot \mathrm{N} < \tau_{stall}$

$\Leftrightarrow$ N<5.102

$N \equiv \frac{r_2}{r_1}$ (gear ratio)

$\tau_{mot} = \tau_{friction} \cdot \frac{r_2}{r_1} = \tau_{friction} \cdot \mathrm{N}$

2. Motor temperature is lower than 35℃

(using heat equation)

Time= 2min, safety factor=4.0

N<2.829

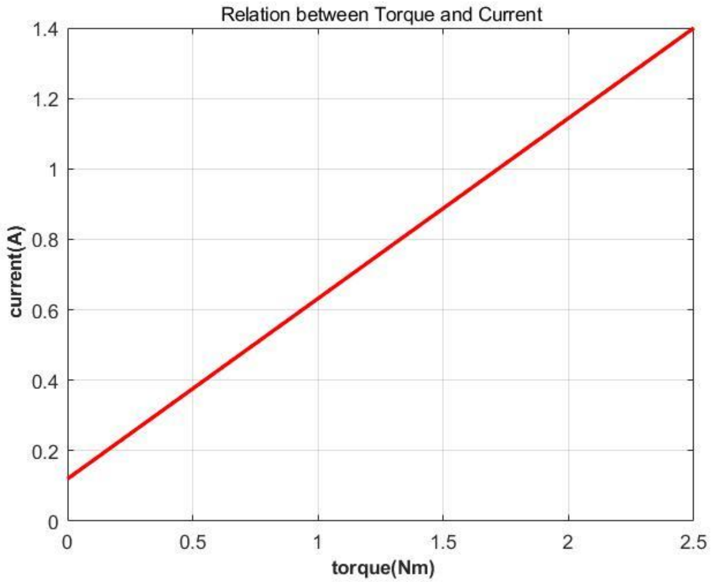
Experimental data

ω_{mot} = 37.375 rpm

I = 0.4 A

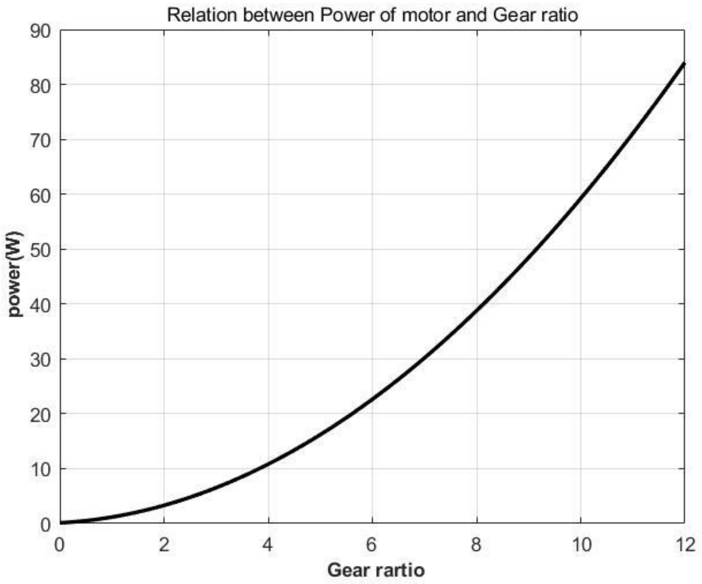
$\tau_{friction}$ = 0.49 N· m

R = 8.5714 V/A



I= $0.12 - 0.512\tau_{mot}(A)$

$P = I^2 R = 0.5395N^2 + 0.5161N + 0.1234$ (J/s)



When $\tau > 0.49$,
Efficiency : 0.08~0.38
Worst case : 0.08

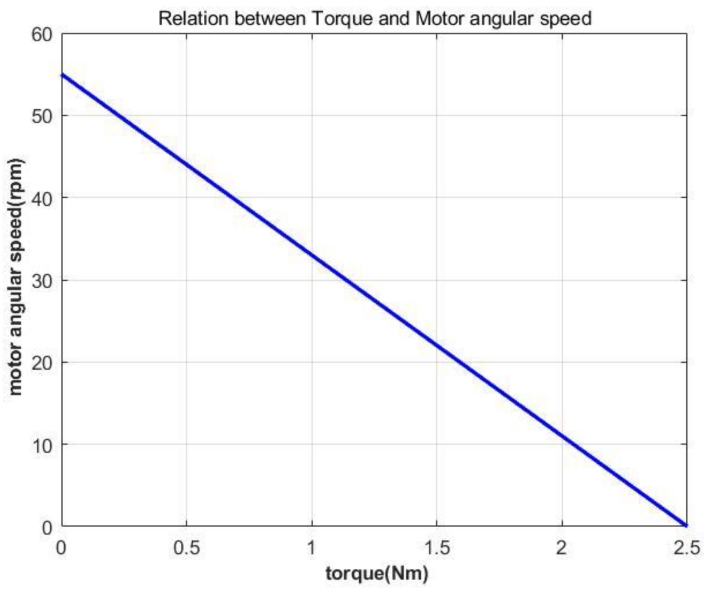
$P \cdot 0.92 \cdot t = R_{ther} \Delta T$

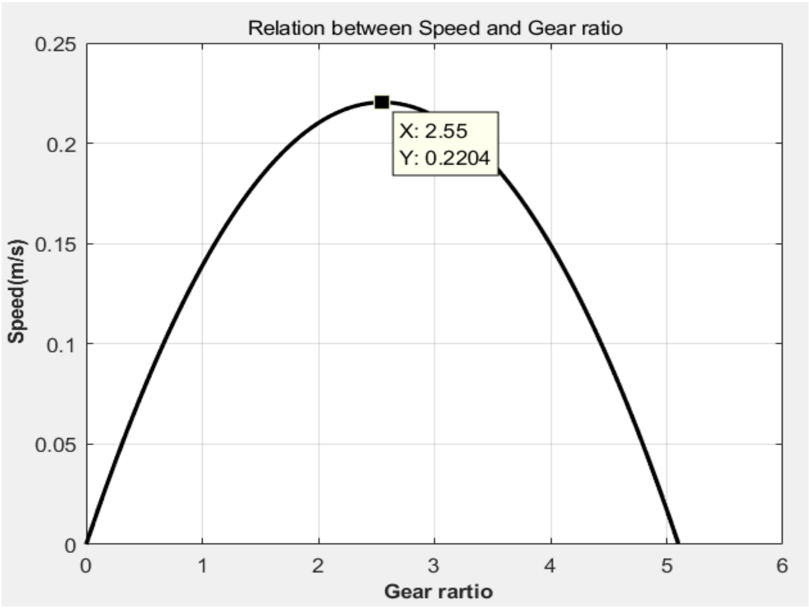
Let $T_0 = 26$℃, $\Delta T < 9$ ℃
P< 75.04
N<11.315

In experiment

$18R_{ther} = {I_{eff}}^2 \cdot R \cdot 0.62 \cdot 1200$
$R_{ther} = 920.5$ (J/k)

If the safety factor is 4.0 -> N<2.829



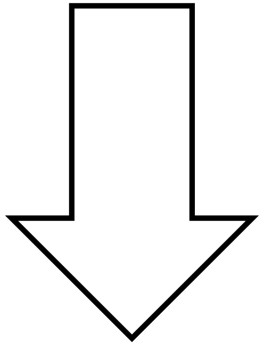


3. Motor speed $\propto X$ (at same angular velocity)
Angular velocity $\propto -X$ $(linearly\ decrease)$
$\therefore$ optimization

$$\omega_{mot} = 5.7596 - 2.3038\tau_{mot}(rad/s)$$

$$\omega_{mot} = 5.7596 - 2.3038 \cdot \tau_{friction} \cdot \mathrm{N}$$

$$V_{speed} = r_{wheel} \cdot \omega_{gear_1} = \omega_{mot} \cdot \frac{r_2}{r_1} = \omega_{mot} \cdot N$$



$$V_{speed} = 0.17278 \cdot \mathrm{N} - 0.00338659 \cdot N^2$$

$$V_{max} \ \ at\ N = 2.551$$