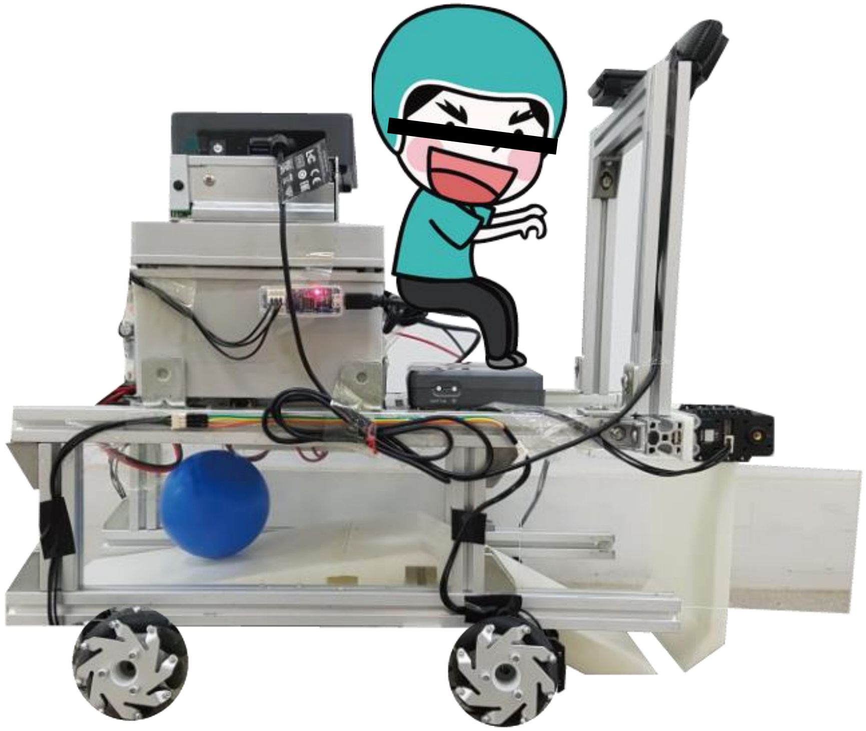
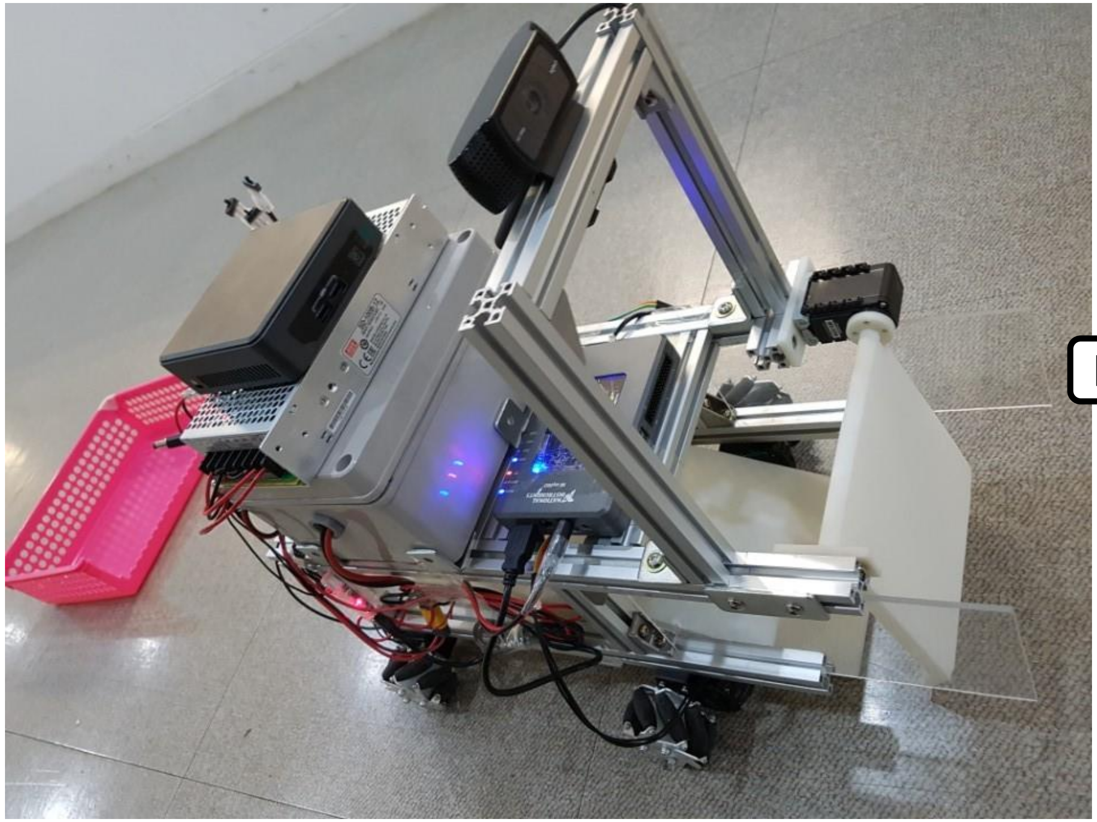
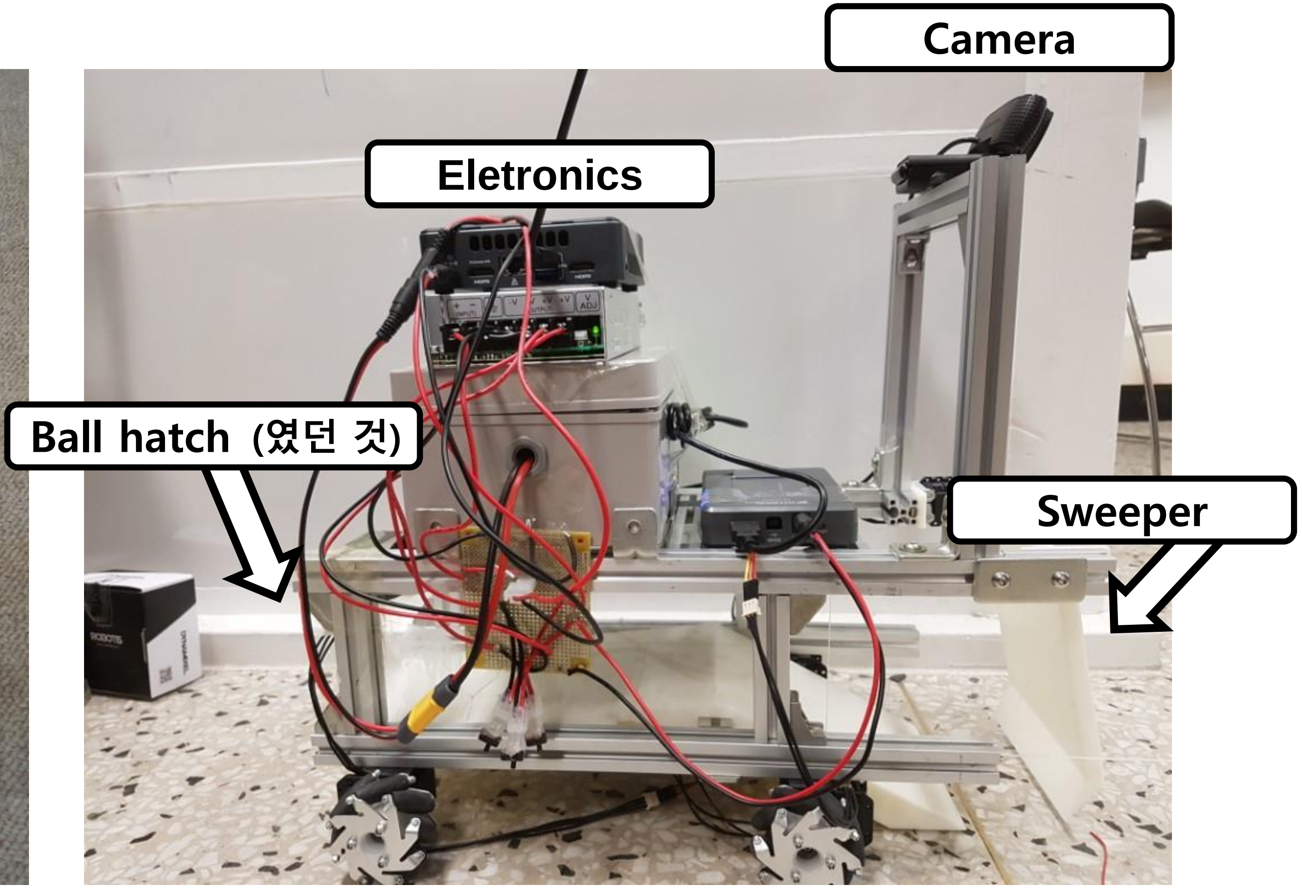


2nd Design Review

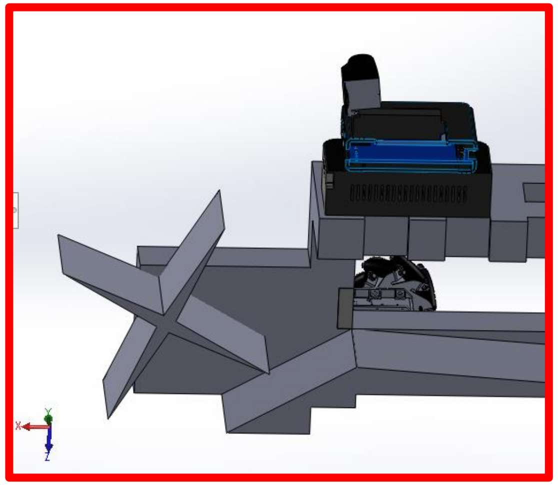
Capstone7 / 배달의민족

Overall Design





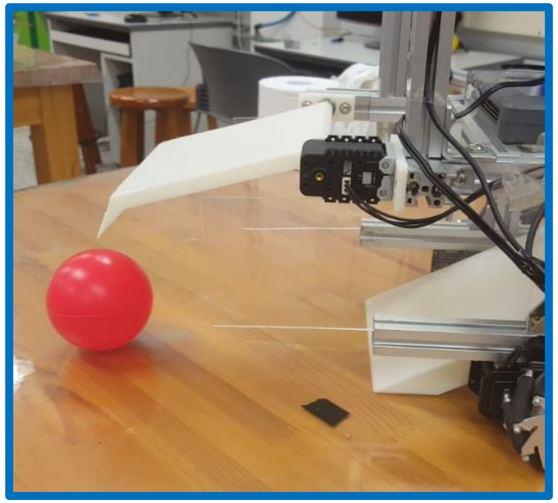
New Sweeper Design



Too big

Motor Itself need more energy

Blade may squeeze the ball



- Reduce to single blade
- Make end to be angled

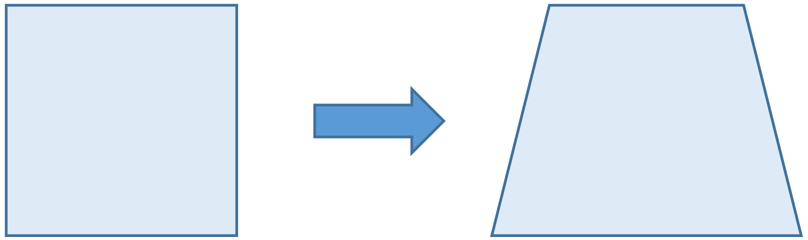
Further Improvement

Major Problem:

1. Overall weight
2. Distribution of the load
3. Stability

We can solve it with…

- Adjust the support (Widened base)



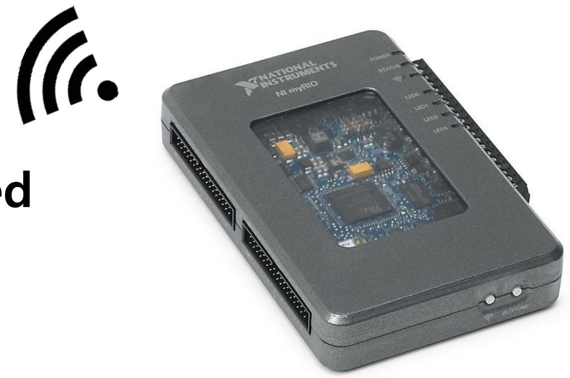
- Additional wheel

LabView

TCP/IP Connection

float[24]

1. Move all direction with controllable speed
2. Rotation with controllable speed
3. Control the sweeper and the rear gate



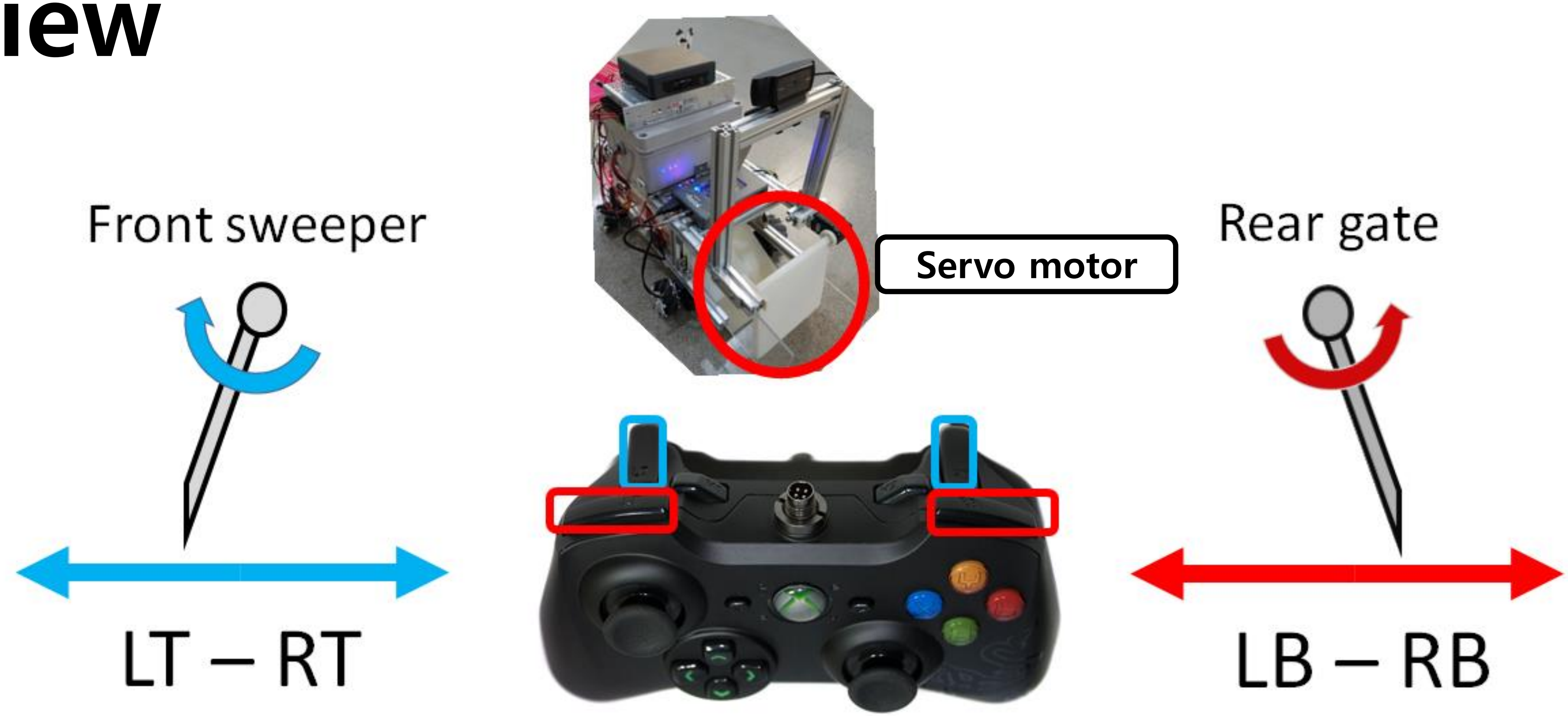
r (magnitude) ∝ Velocity
θ = direction

X-magnitude ∝ rotation

Left stick

Right stick

LabView



Future development

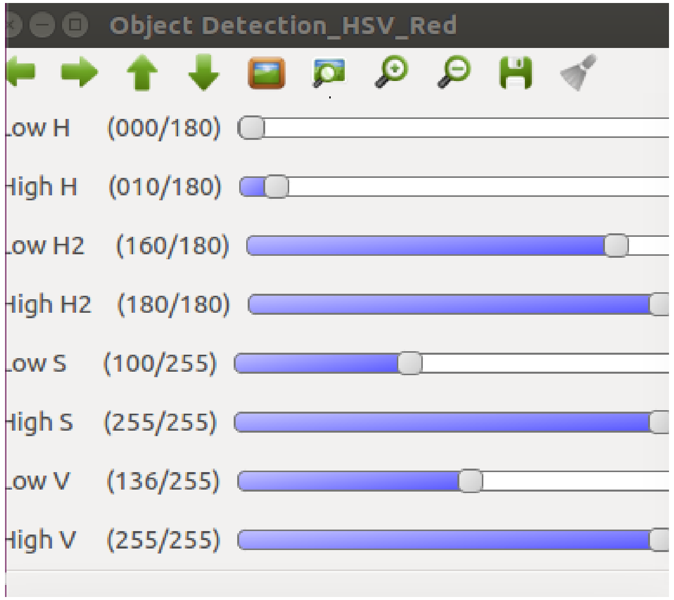
- Find exact relation between the voltage and actual motor speed
- Faster Communication (manipulate the data size)

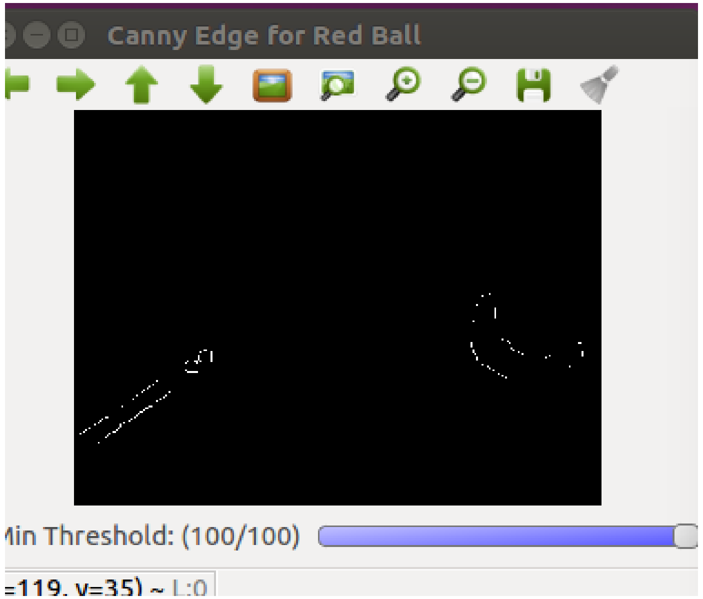
OpenCV

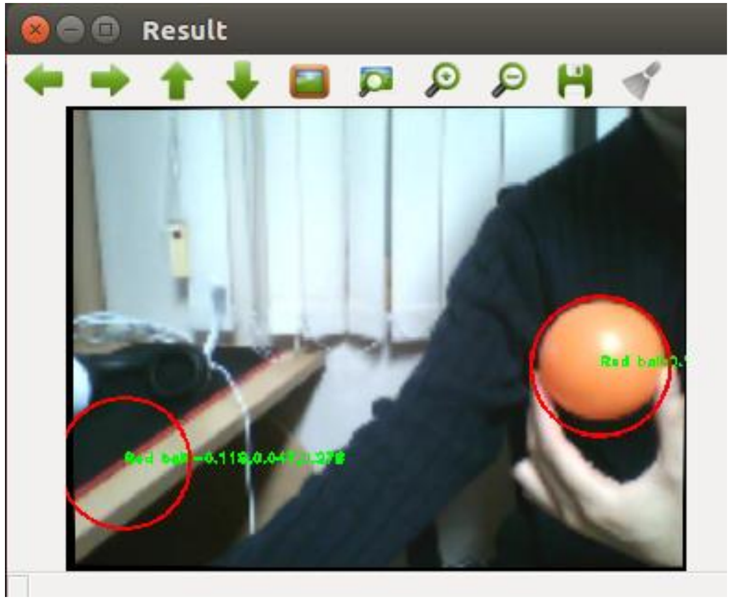
Current Function

-Detect & Locate Ball
-Determine Color
-Reduce Noise

-Camera Calibration
-Use radius of ball to calculate x, y, z distances
-Find Center of Ball





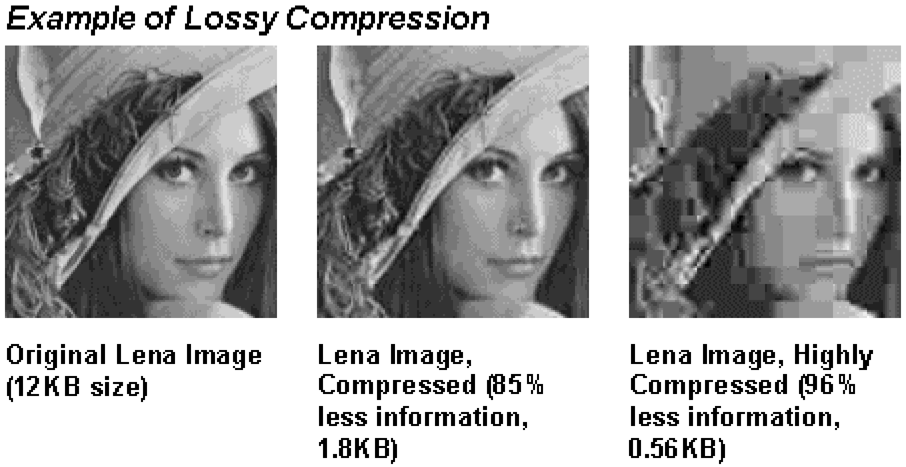


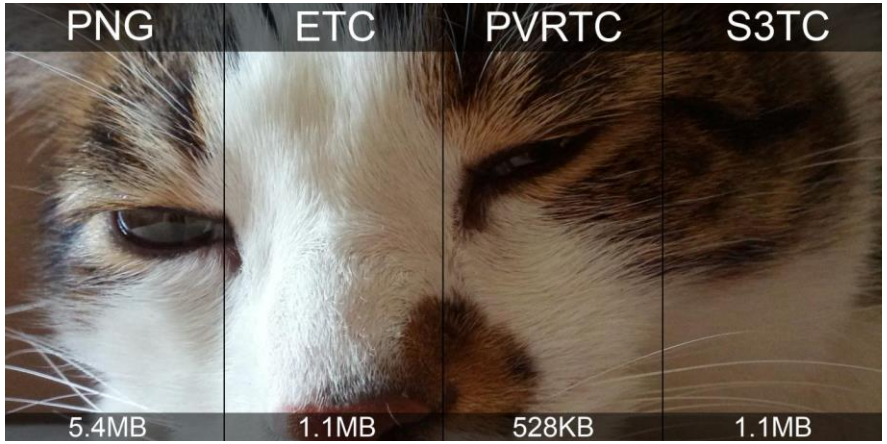
Problem: Narrow FOV / Data Traffic due to Large Data Sets

OpenCV

FOV - lens

Data compression





OpenCV

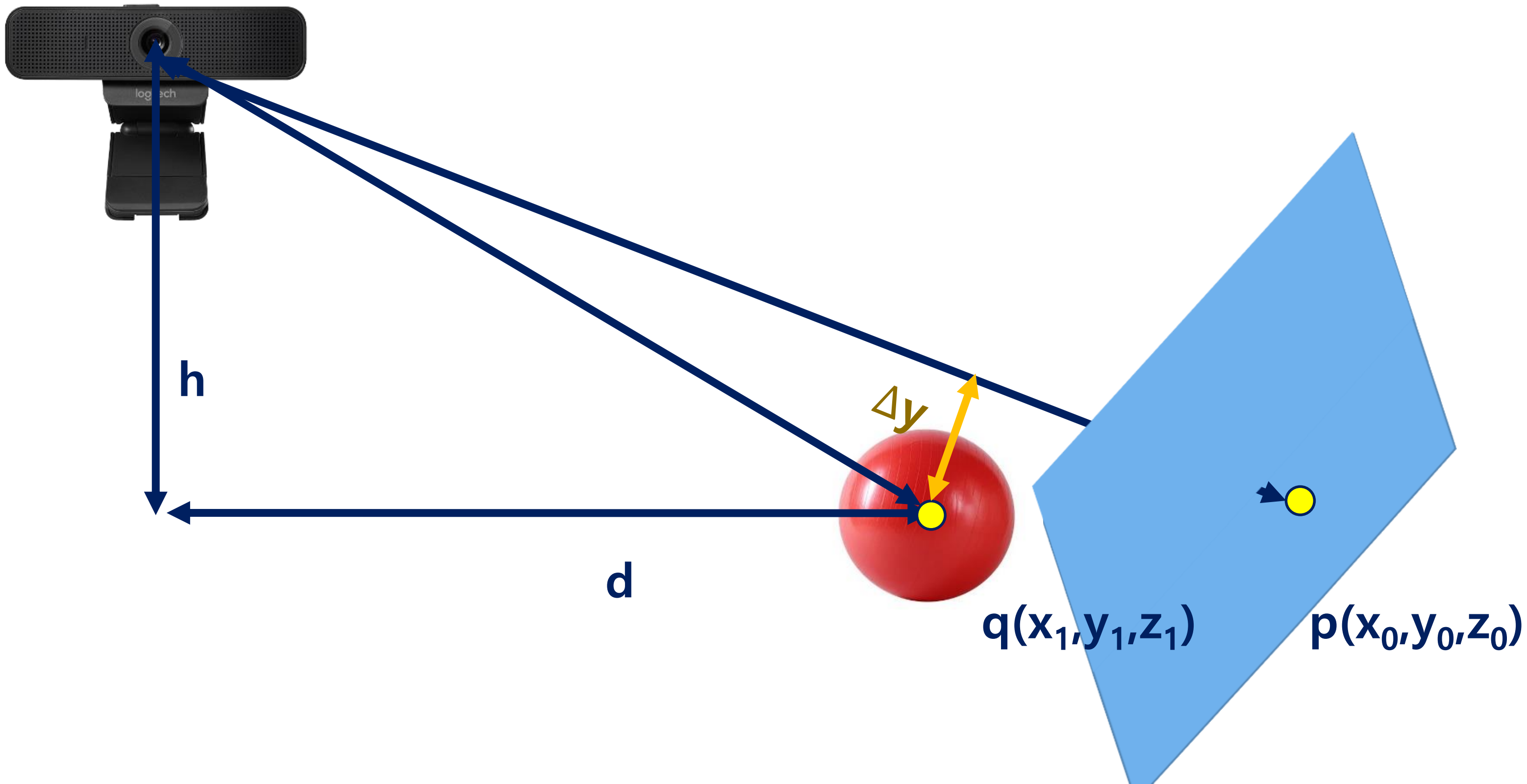
Future Development

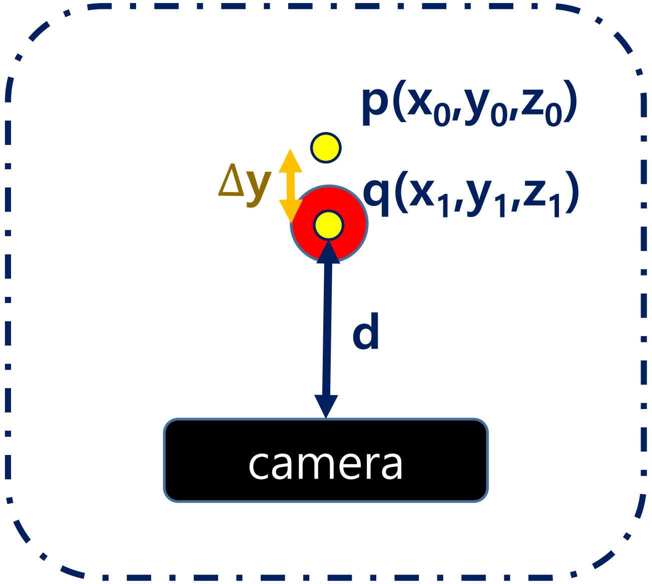
Increase precision of distance calculation

-Use center of ball, rather than radius of ball to calculate distance

-Because center information is more stable

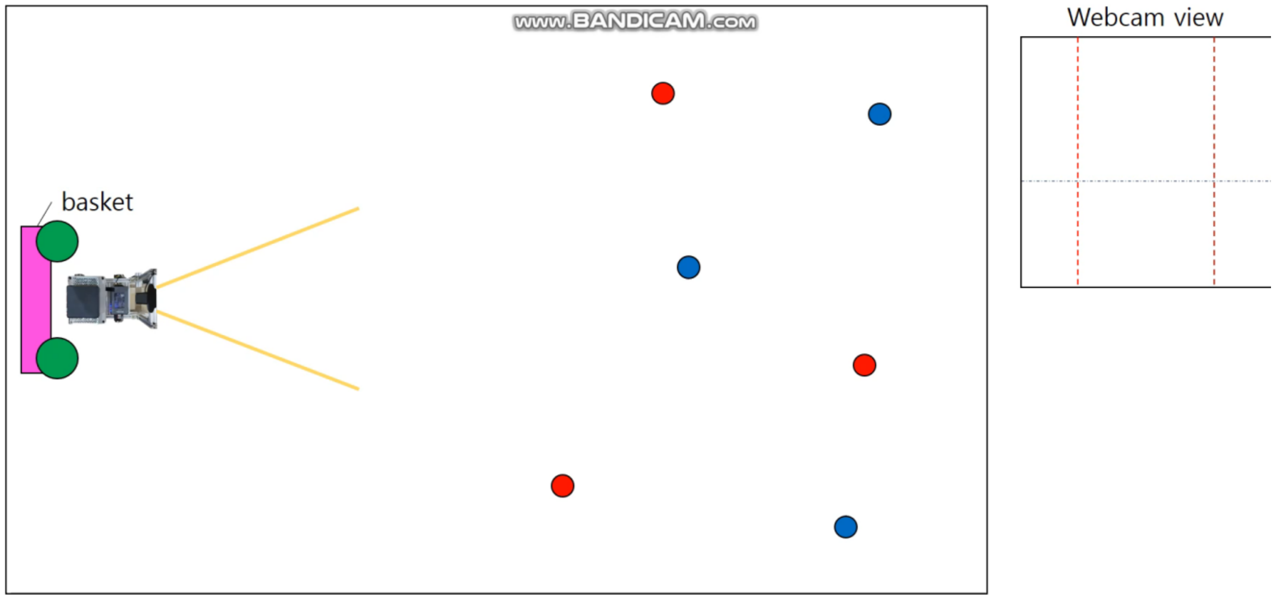
Use pass filter + average values of multiple data to resolve outlier information problem





ROS Integration

Underlying Algorithm



Case 1 : Default

Condition : B.D = NULL || #BALL < 3

Random walk

Rotate once at midpoint / specific point

Case 2 : Catching

Condition : B.D != NULL || #BALL < 3

Ball Detection information

(1) Orient (2) Go Pick up

Case 3 : Go Back

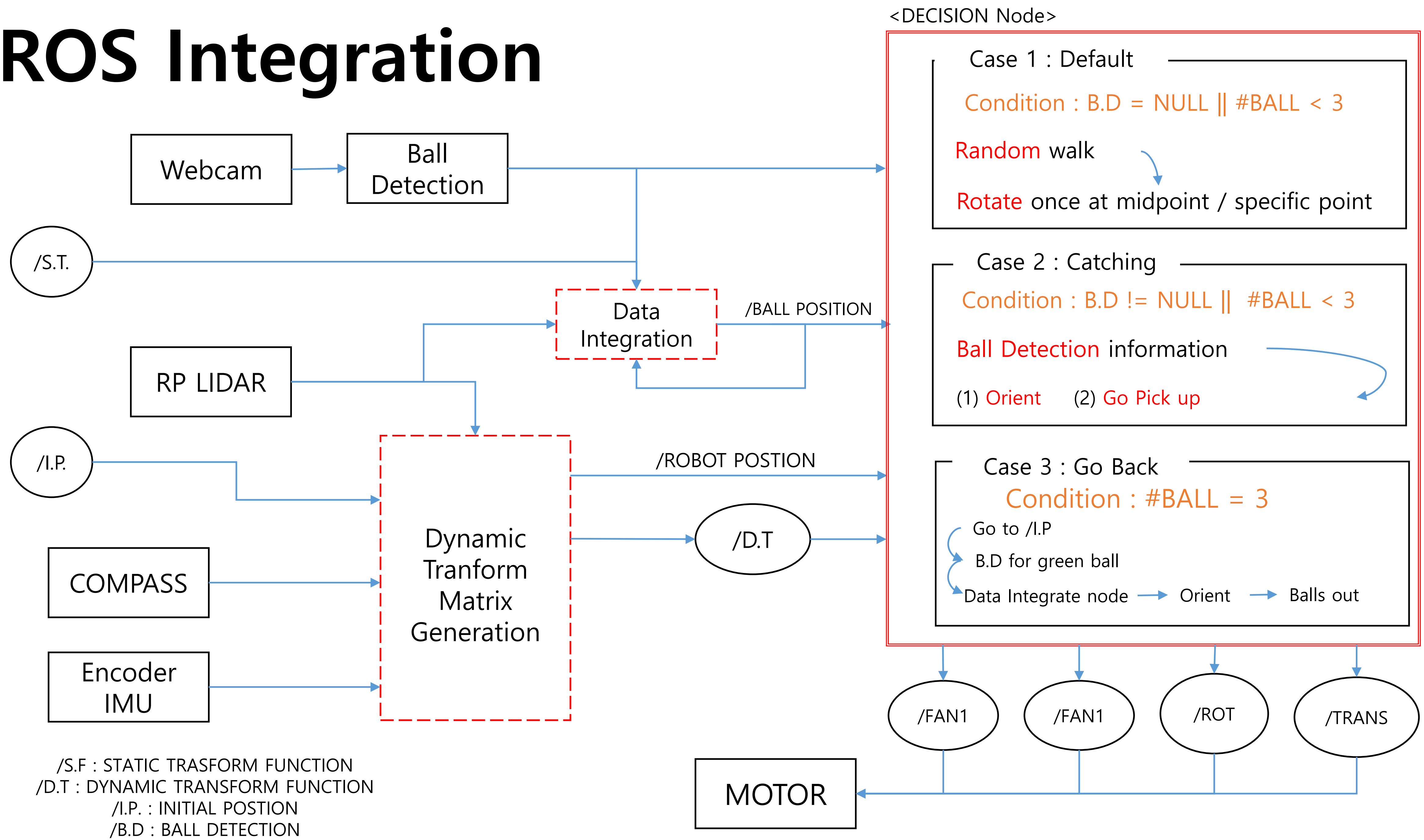
Condition : #BALL = 3

Go to /I.P

B.D for green ball

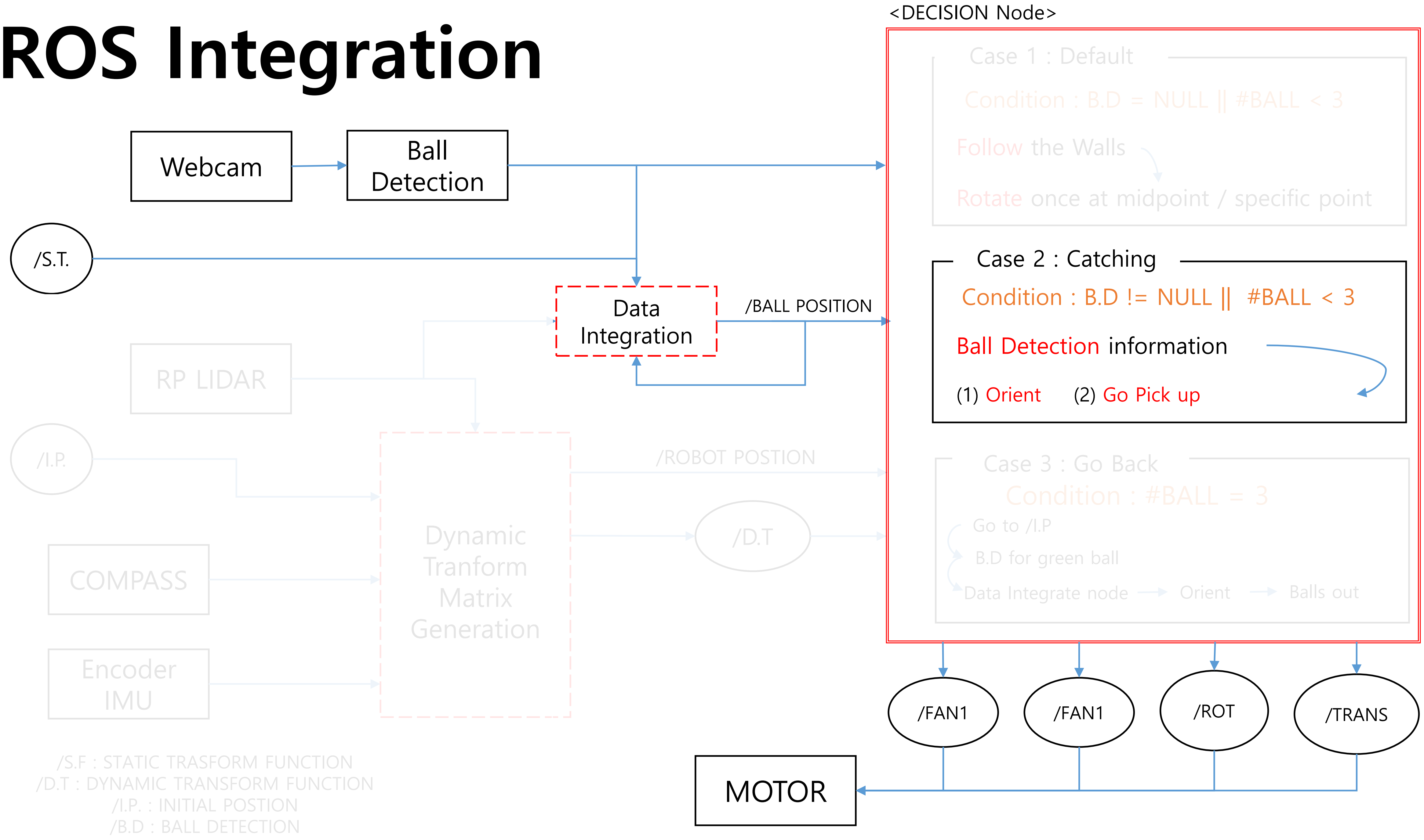
Data Integrate node → Orient → Balls out

ROS Integration

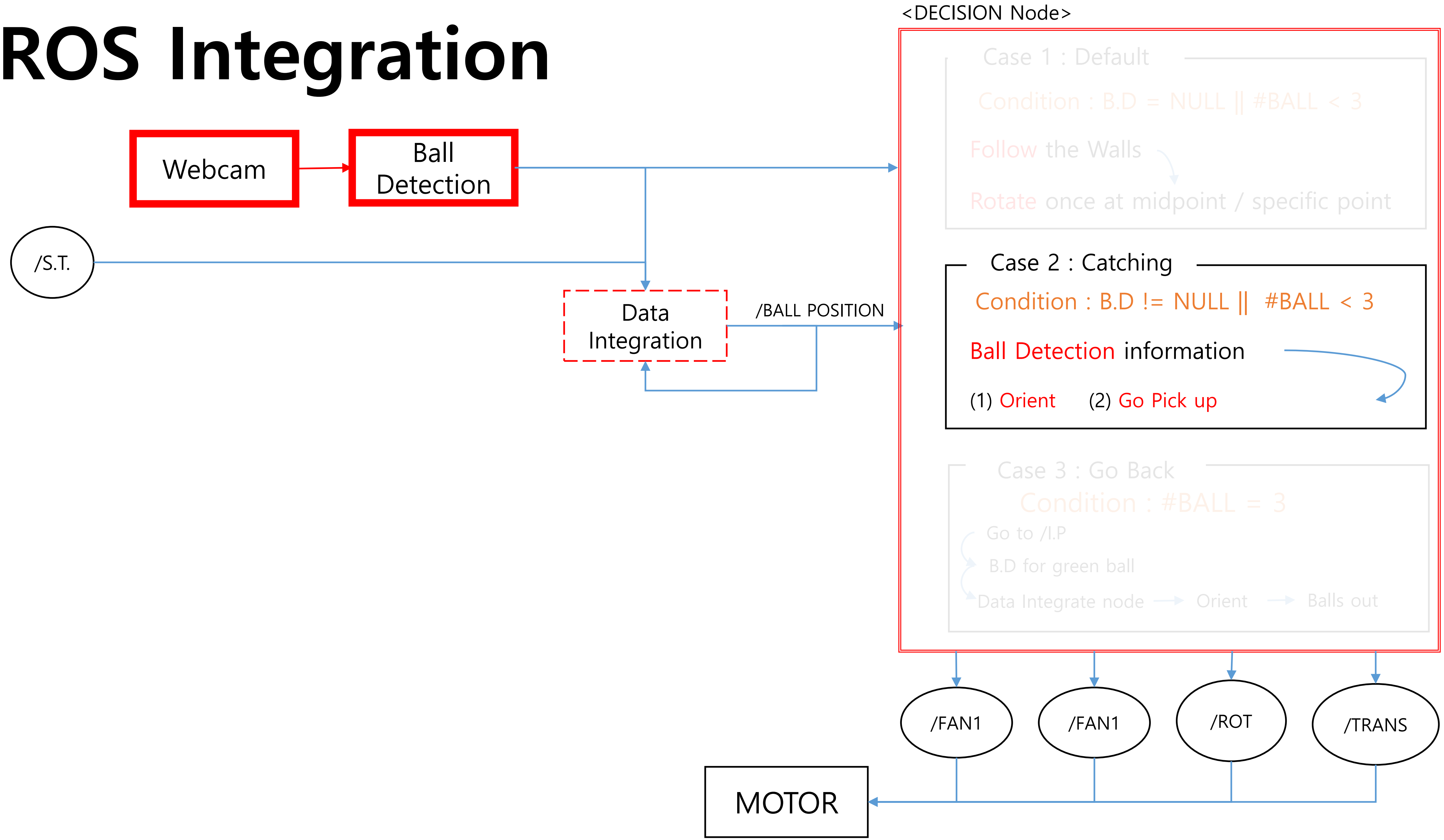


/S.F : STATIC TRASFORM FUNCTION
/D.T : DYNAMIC TRANSFORM FUNCTION
/I.P. : INITIAL POSTION
/B.D : BALL DETECTION

ROS Integration



ROS Integration



ROS Integration

Webcam → Ball Detection

<DECISION Node>

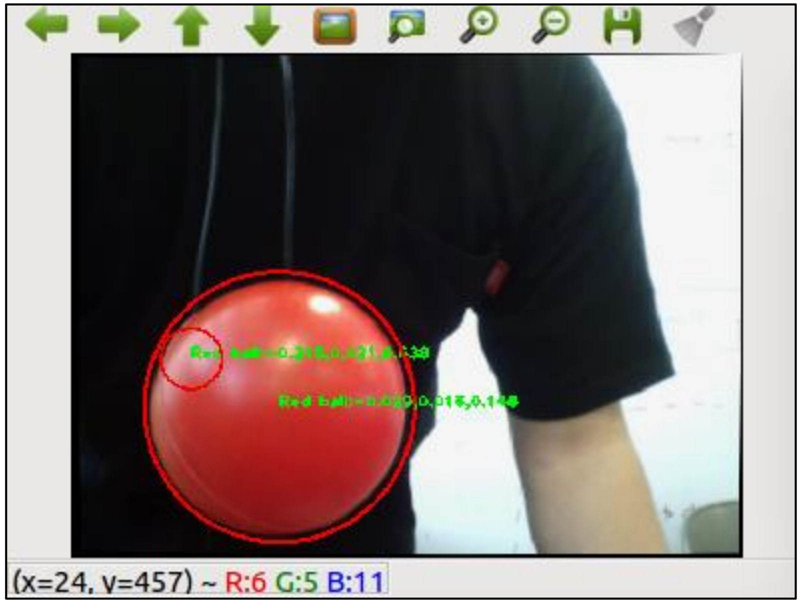
Case 1 : Default

Condition : B.D = NULL || #BALL < 3

Follow the Walls

Rotate once at midpoint / specific point

/S.T.

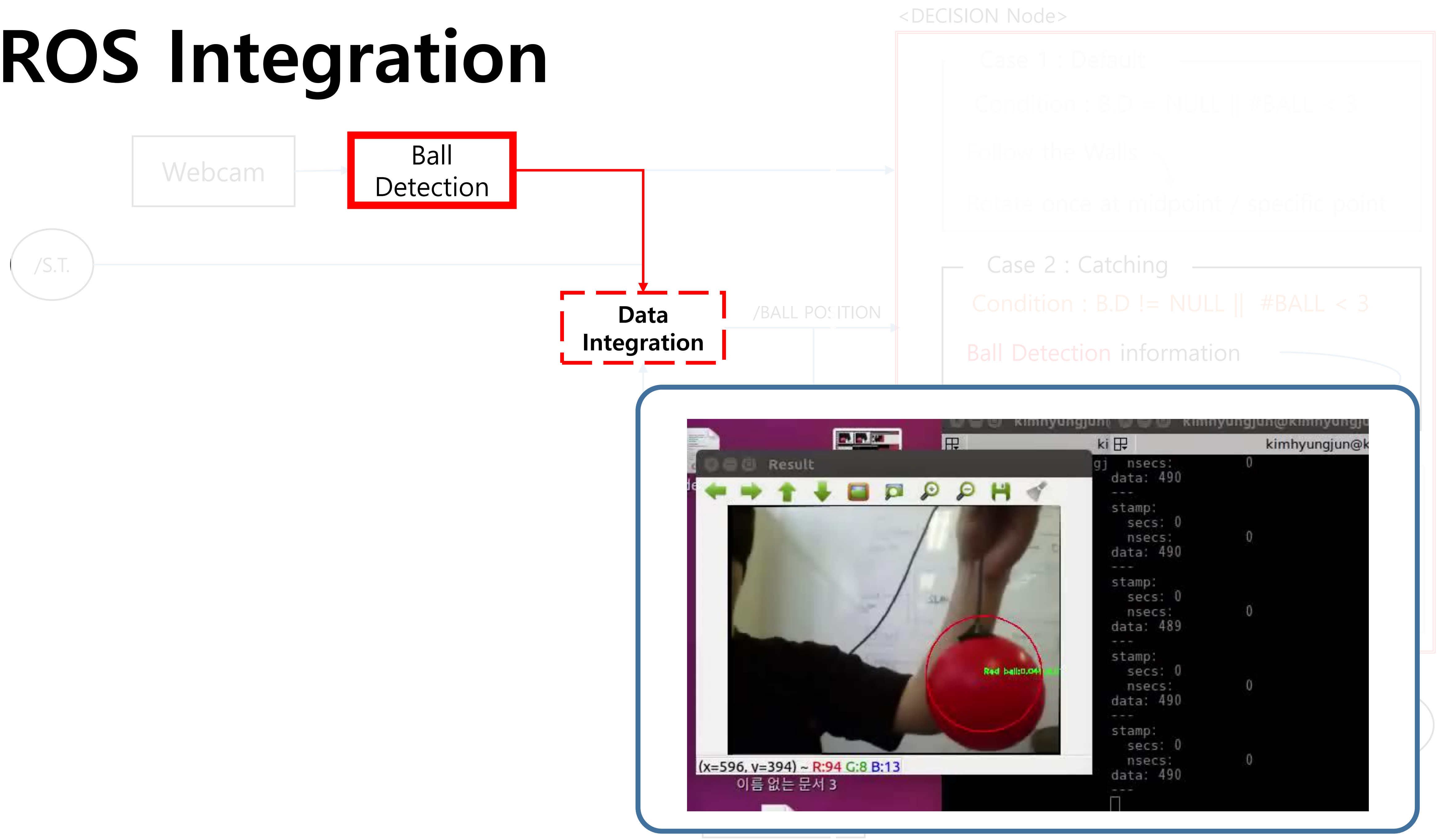


Many balls are detected
&
Many zero data

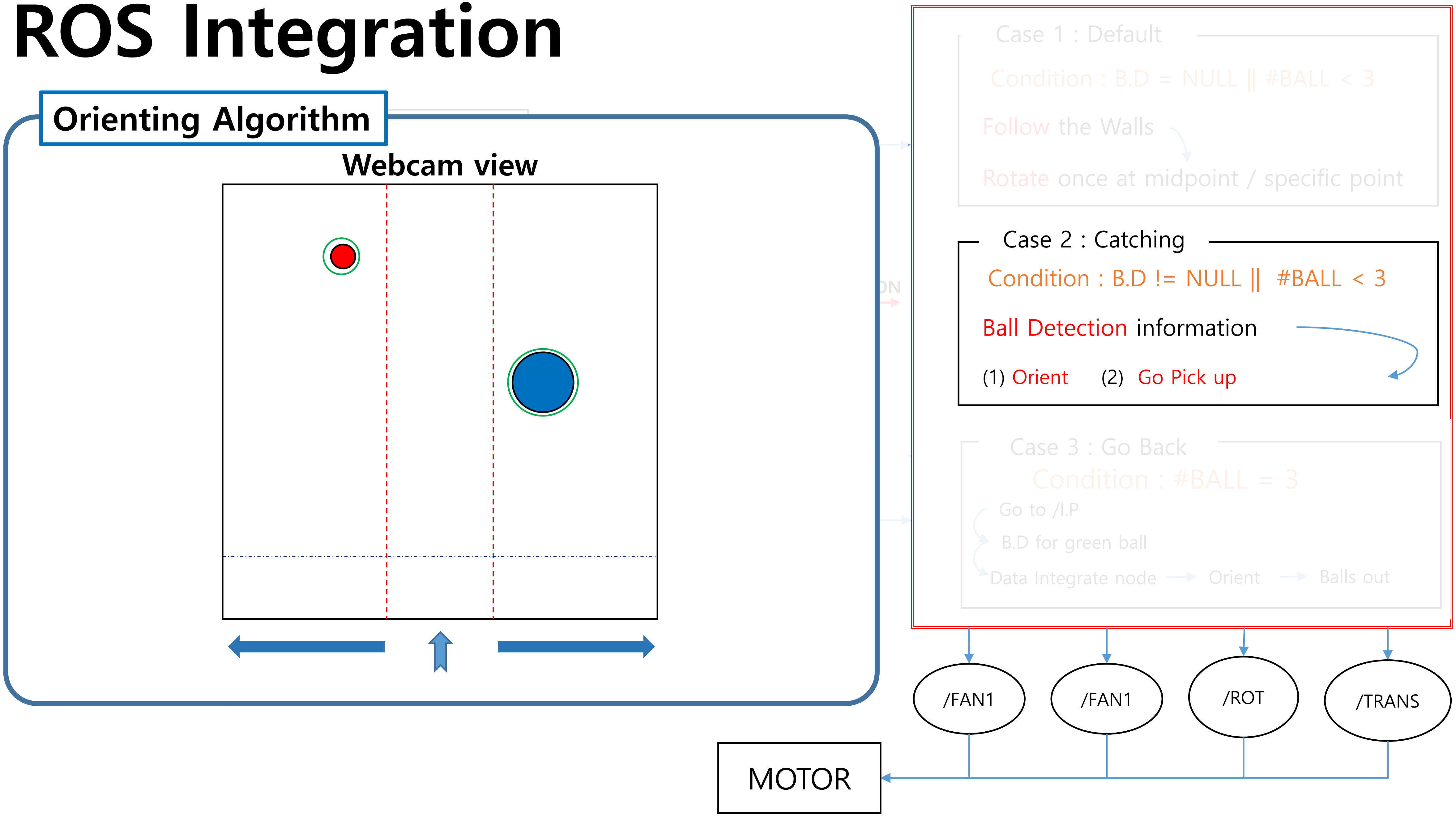
➡

Which data should be use?

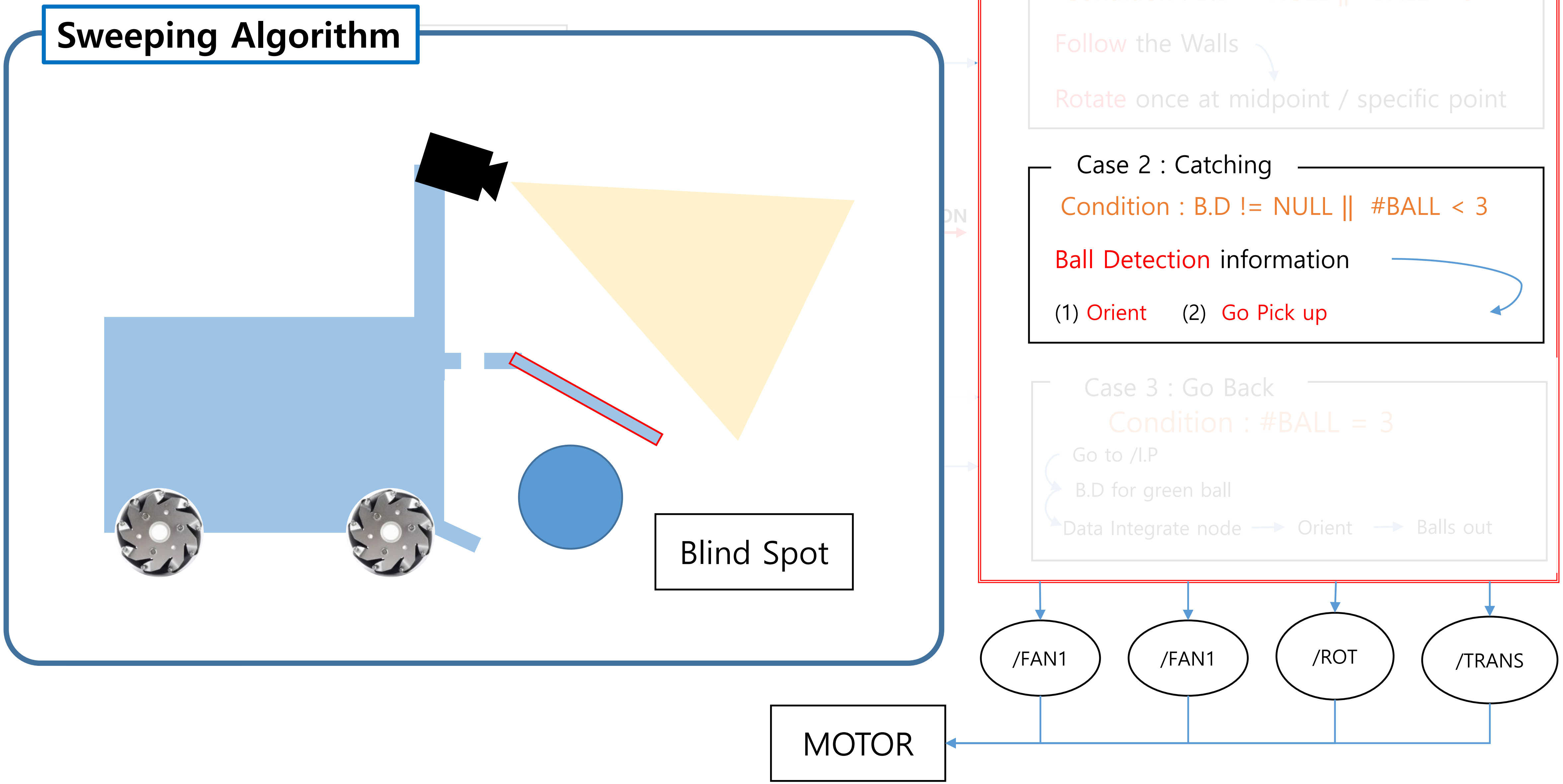
ROS Integration



ROS Integration

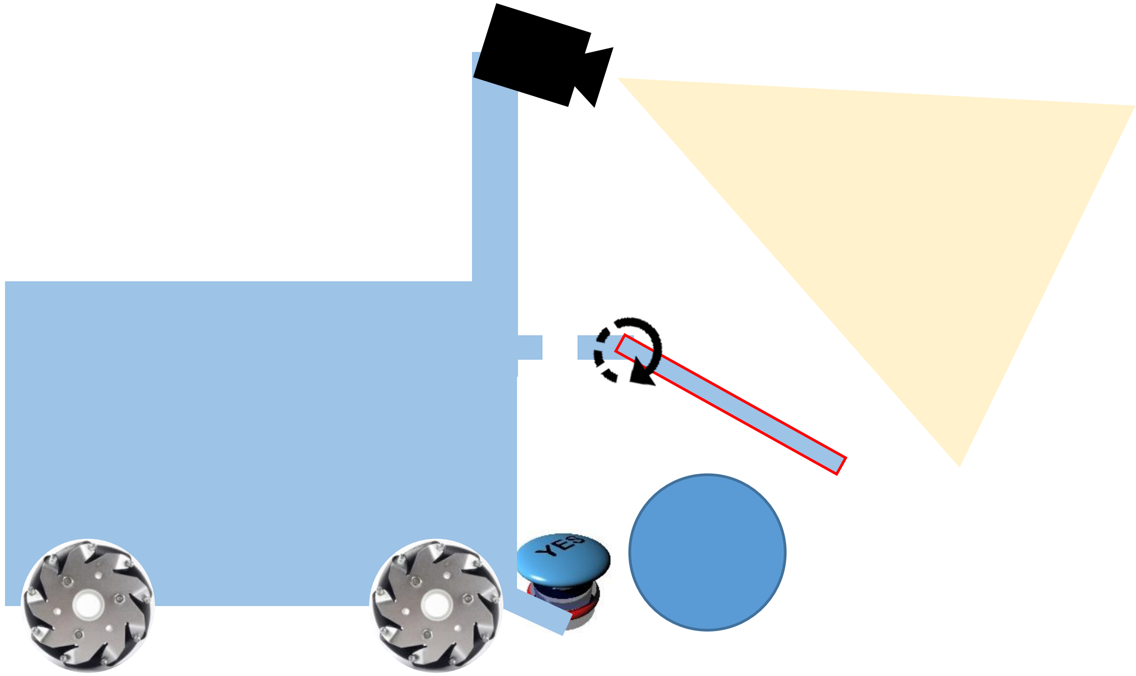


ROS Integration



ROS Integration

Sweeping Algorithm



Case 1 : Default

Condition : B.D = NULL || #BALL < 3

Follow the Walls

Rotate once at midpoint / specific point

Case 2 : Catching

Condition : B.D != NULL || #BALL < 3

Ball Detection information

(1) Orient (2) Go Pick up

Case 3 : Go Back

Condition : #BALL = 3

Go to /I.P

B.D for green ball

Data Integrate node → Orient → Balls out

/FAN1

/FAN1

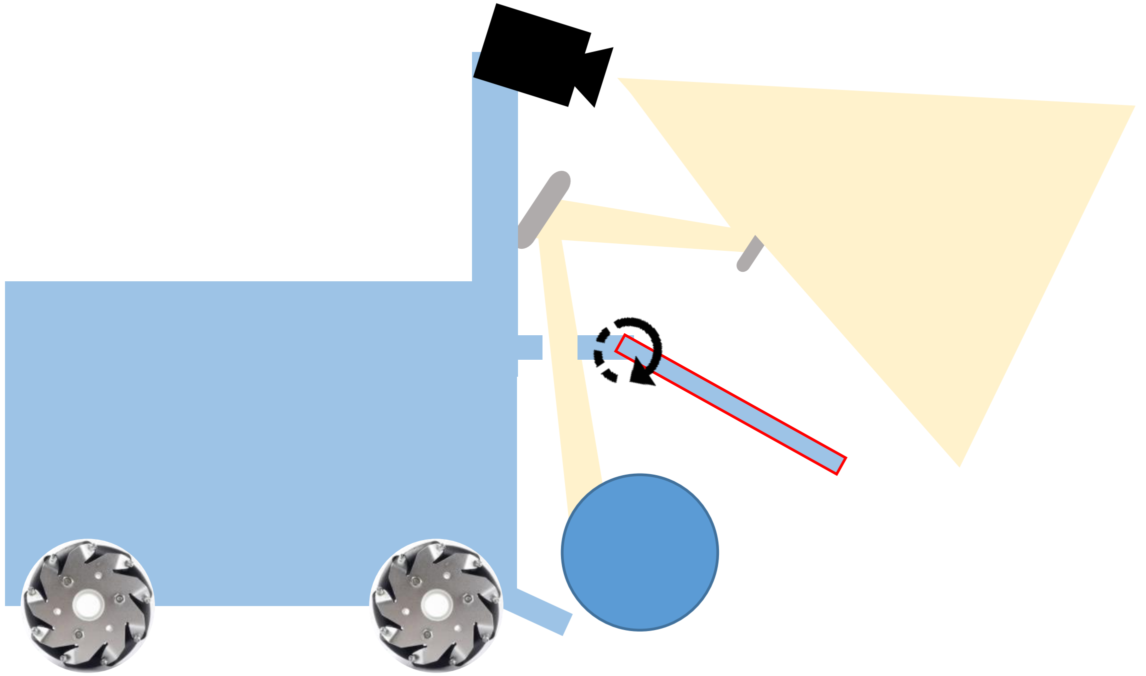
/ROT

/TRANS

MOTOR

ROS Integration

Sweeping Algorithm



Case 1 : Default

Condition : B.D = NULL || #BALL < 3

Follow the Walls

Rotate once at midpoint / specific point

Case 2 : Catching

Condition : B.D != NULL || #BALL < 3

Ball Detection information

(1) Orient (2) Go Pick up

Case 3 : Go Back

Condition : #BALL = 3

Go to /I.P

B.D for green ball

Data Integrate node → Orient → Balls out

/FAN1

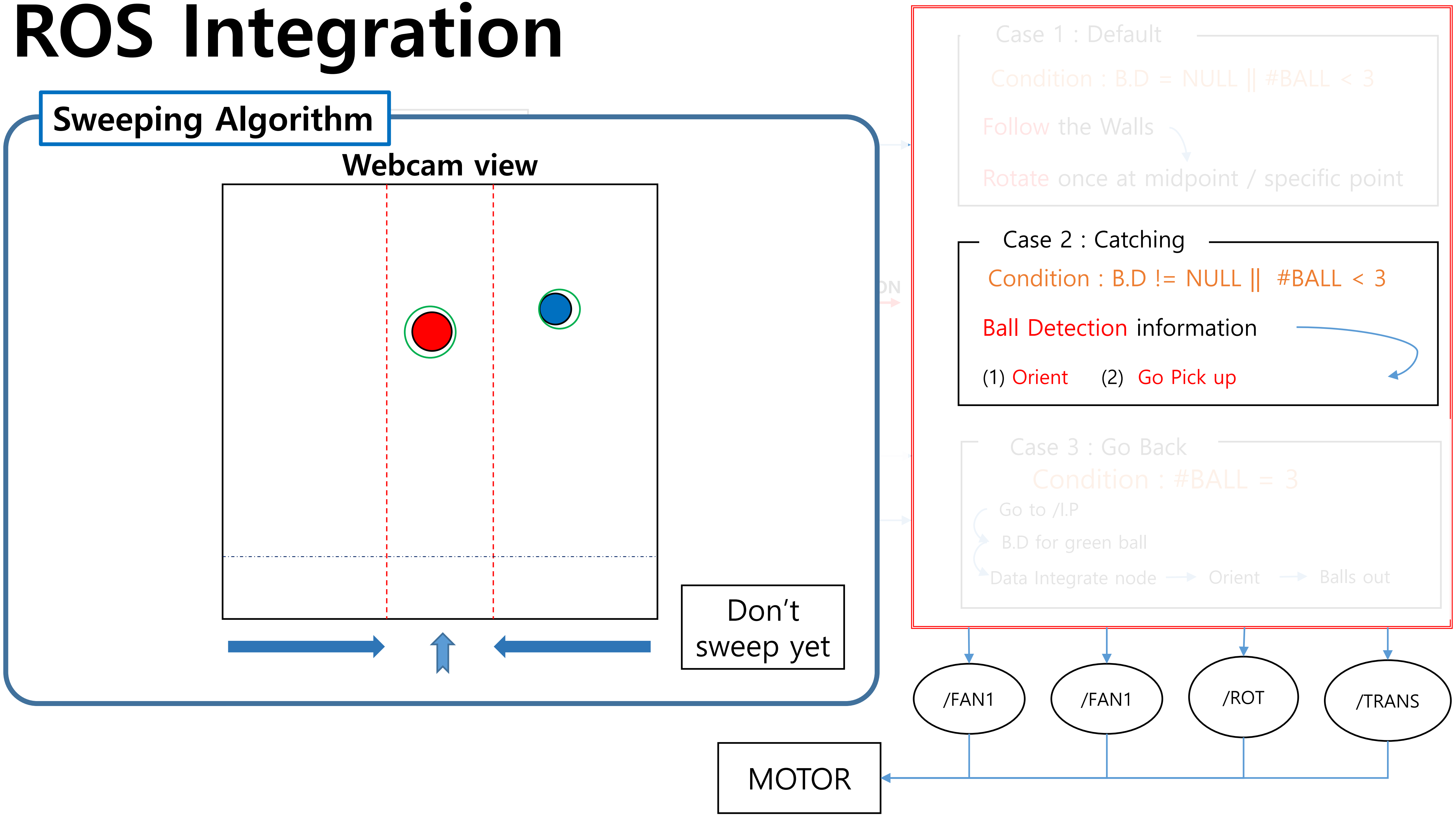
/FAN1

/ROT

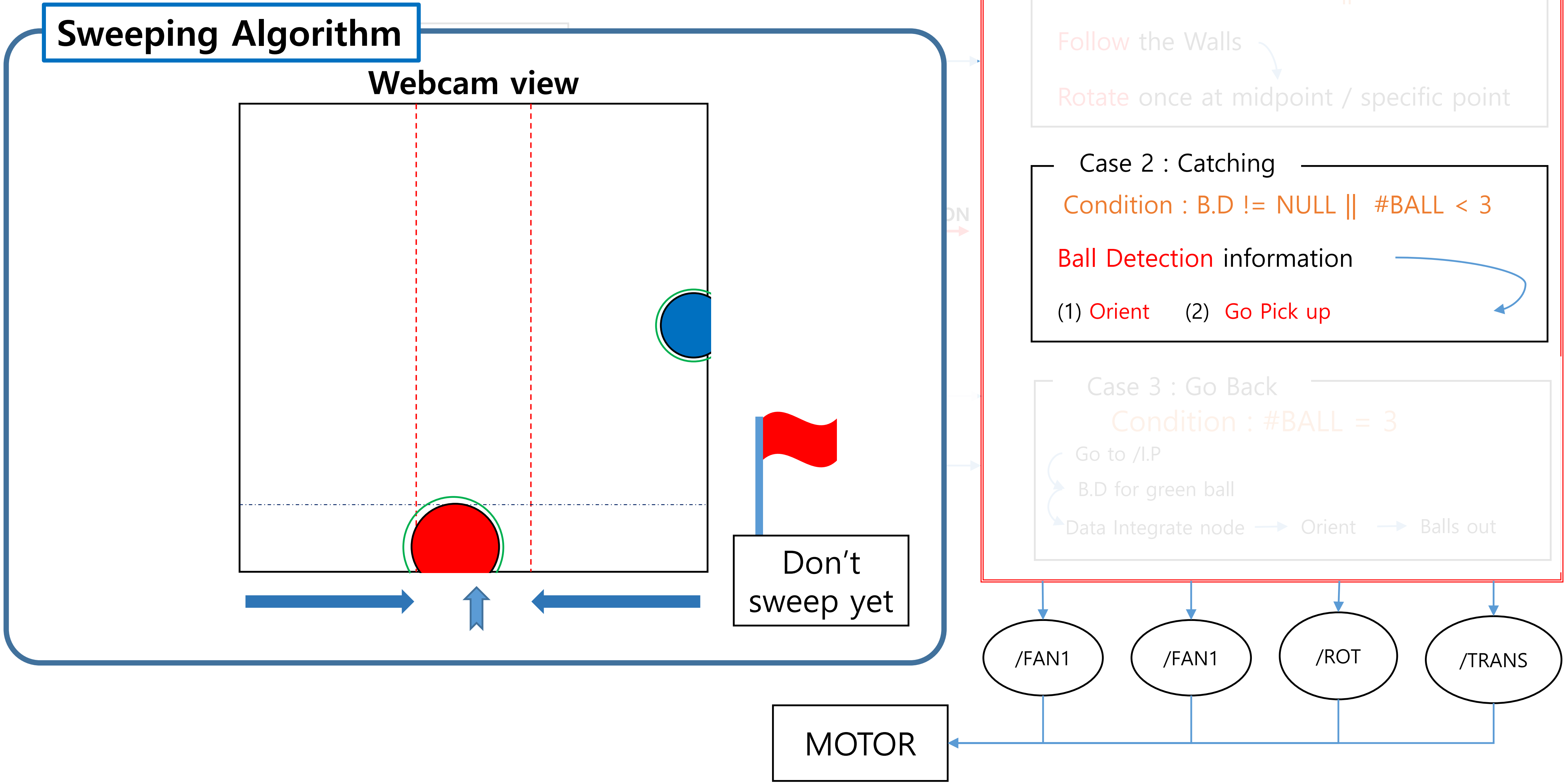
/TRANS

MOTOR

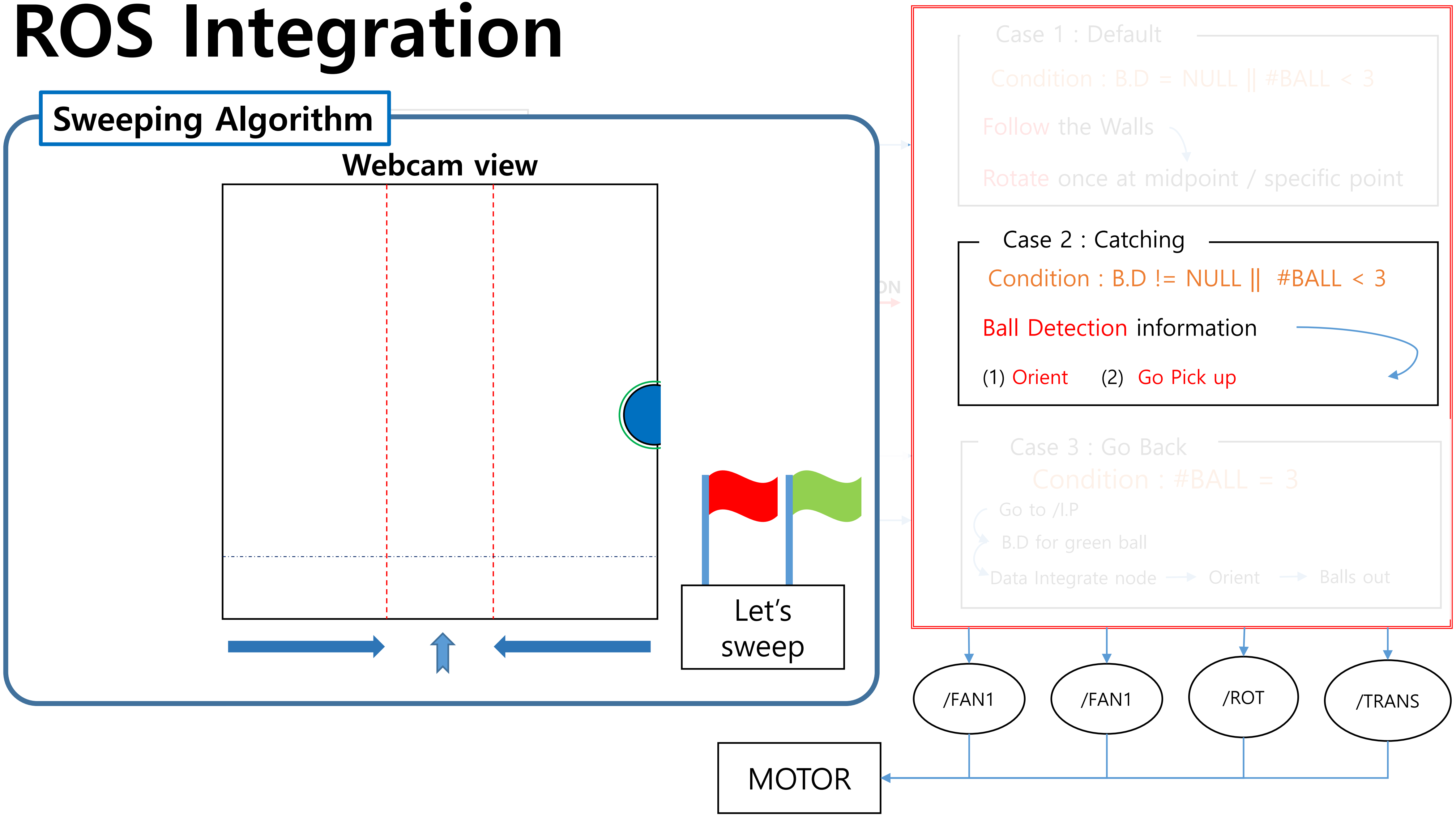
ROS Integration



ROS Integration



ROS Integration



Energy Management / (Vibration)

Least Energy consumption

Mechanical Energy (Main motor) $60W$

+

Electrical Energy $50W$

Intel Nuc : 25W (idle)

NI myRIO : 14W

RPLidar, DFR0315 : 4W

Logitech HD pro webcam :3W

20% of energy heats up battery

$$q_{battery} = hA(T - T_\infty)$$

$$h \sim 20W/Km^2$$

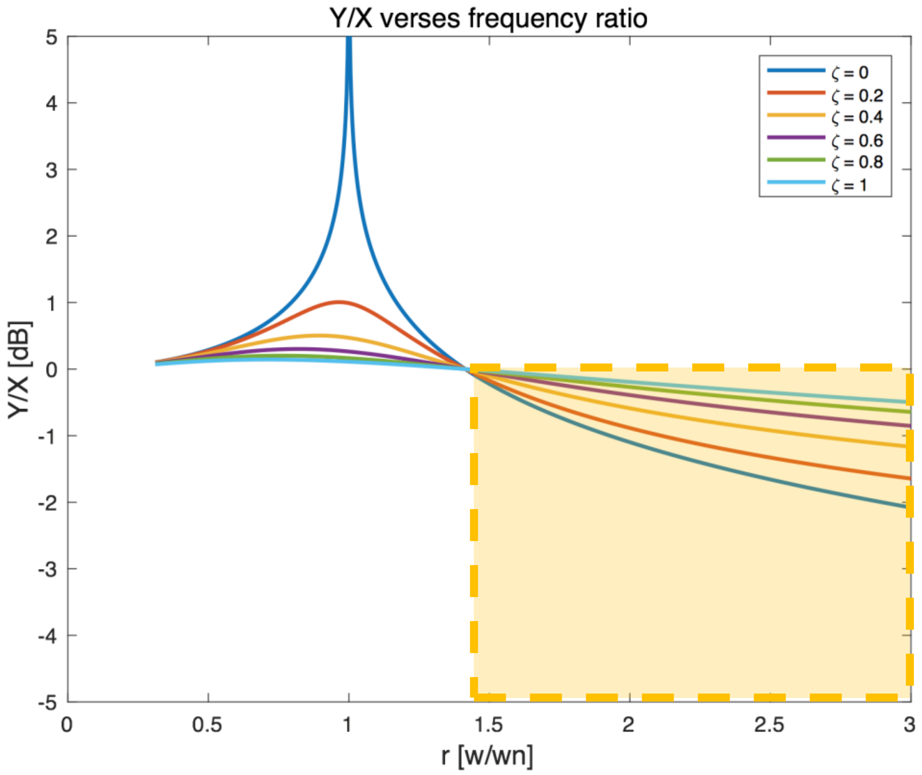
$$A \sim 0.1m^2$$

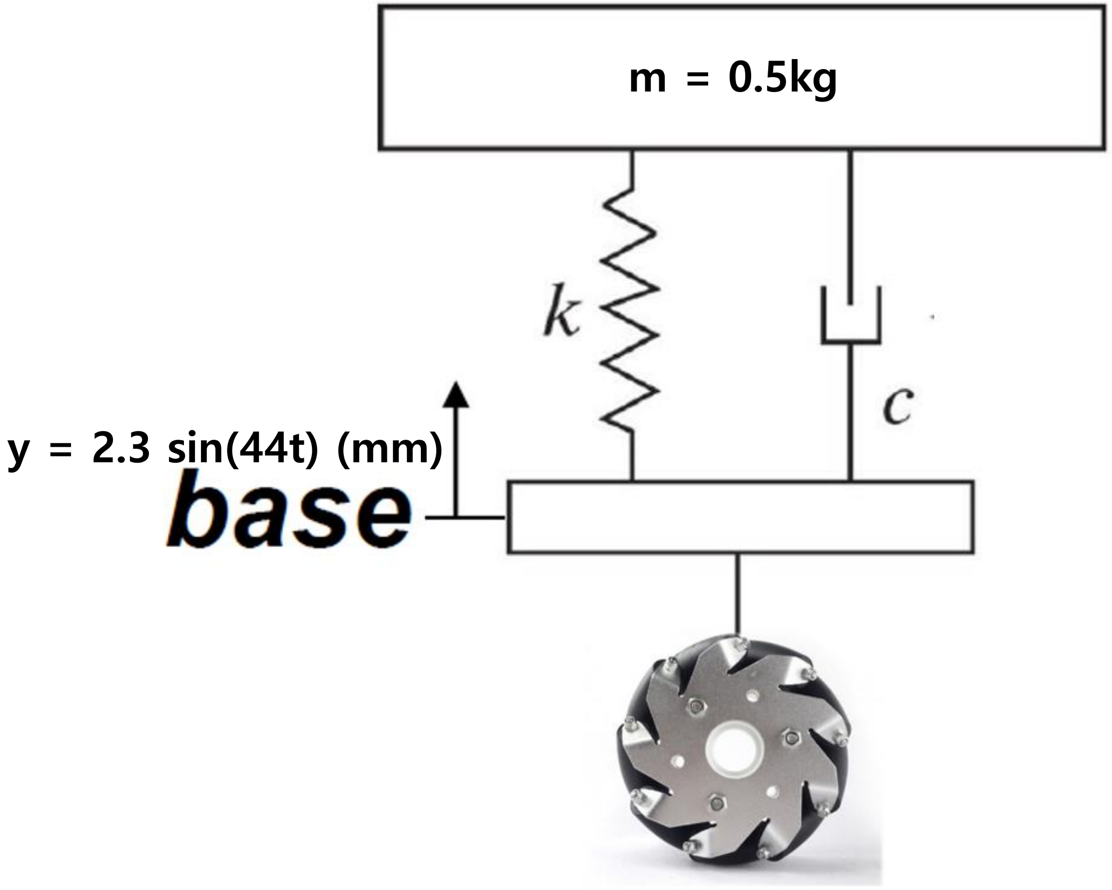
$(T - T_\infty)$ ~ 10K

80% efficiency assumed

Vibration Control

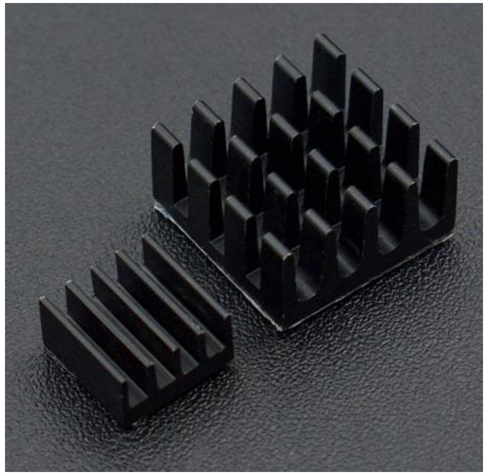
Naïve analysis

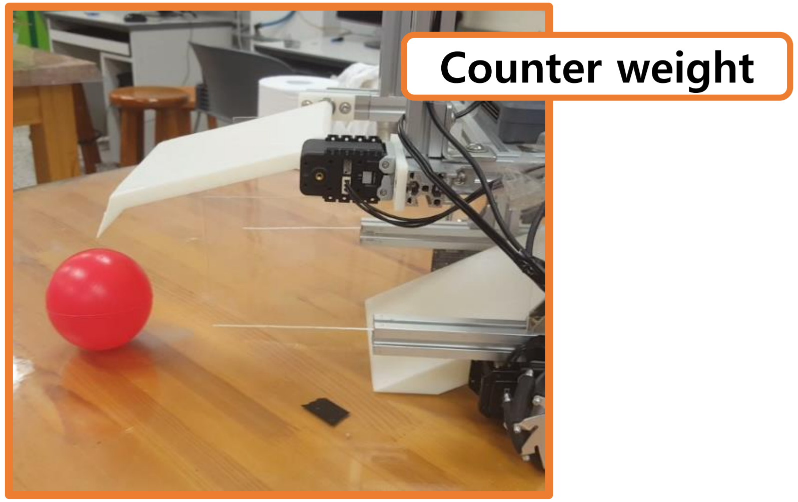




Energy Management / (Vibration)

Major Heat generation: Motor





Demonstration

Manual control / Autonomous