**ME400: 창의적 시스템 구현 - 최종 보고서**

2018-spring

**Team: 걷지 말고 기어**

**20150813 홍성웅**

**§목차**

**1. Hardware Design**

**1.1 Pick-up module**

**1.2 Fin & Fan**

**1.3 Gear-Wheel System**

**1.4 차체**

**2. Software design**

**2.1 Overall algorithm**

**2.2 Ball detection**

**2.3 Pick the ball**

**2.4 go to the basket**

**3. Process**

**3.1 Monthly plan**

**4. Progress Report**

**1. Hardware Design**

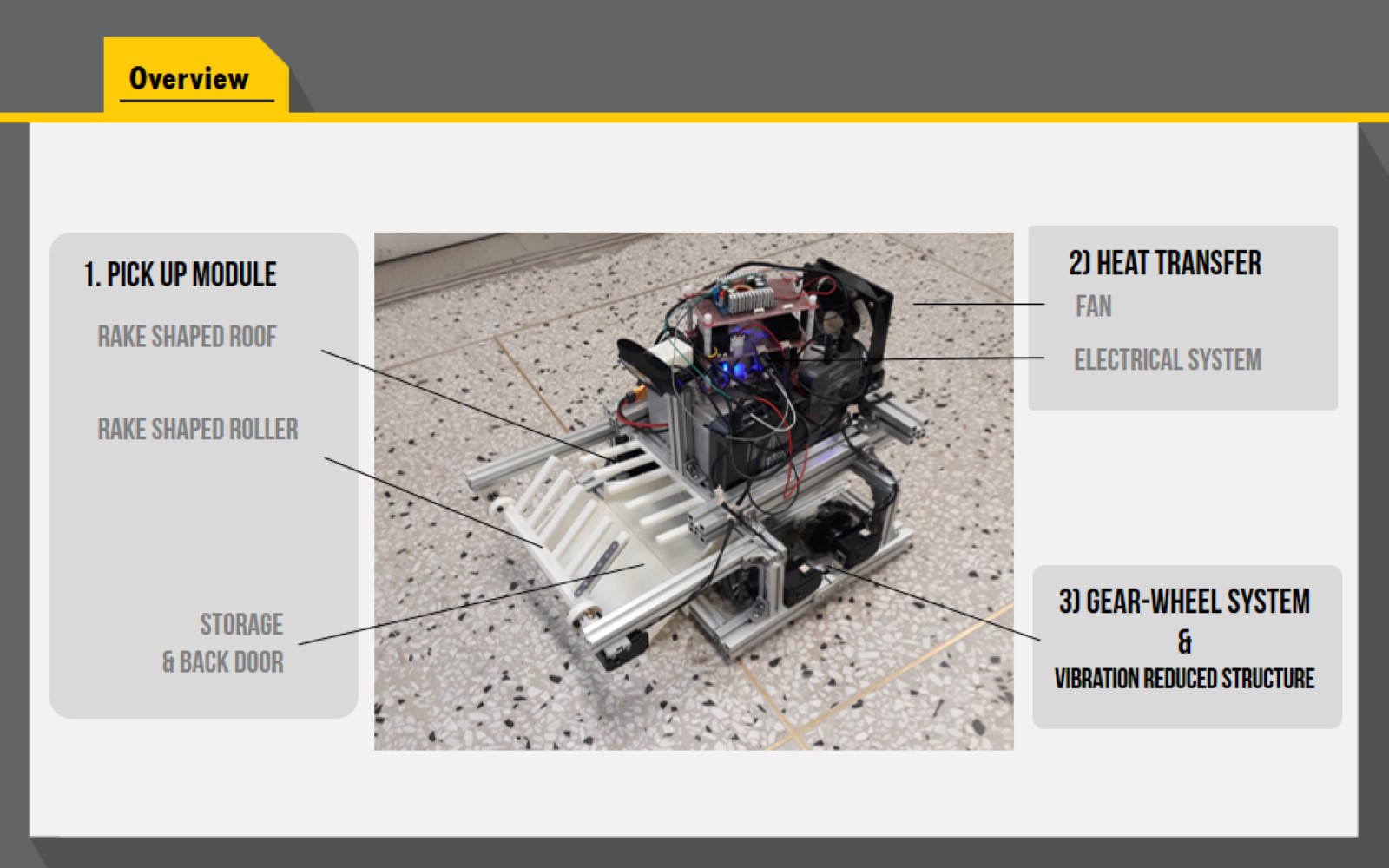
****

Figure 1 Overview of hardware

Mobile platform의 구조는 위와 같다. 특징적인 구조는 pick-up module, 열 전달 설계(fin & fan), 그리고 기어 시스템이 있다. 카메라 위치, 차체의 크기 또한 최적으로 설계하였다.

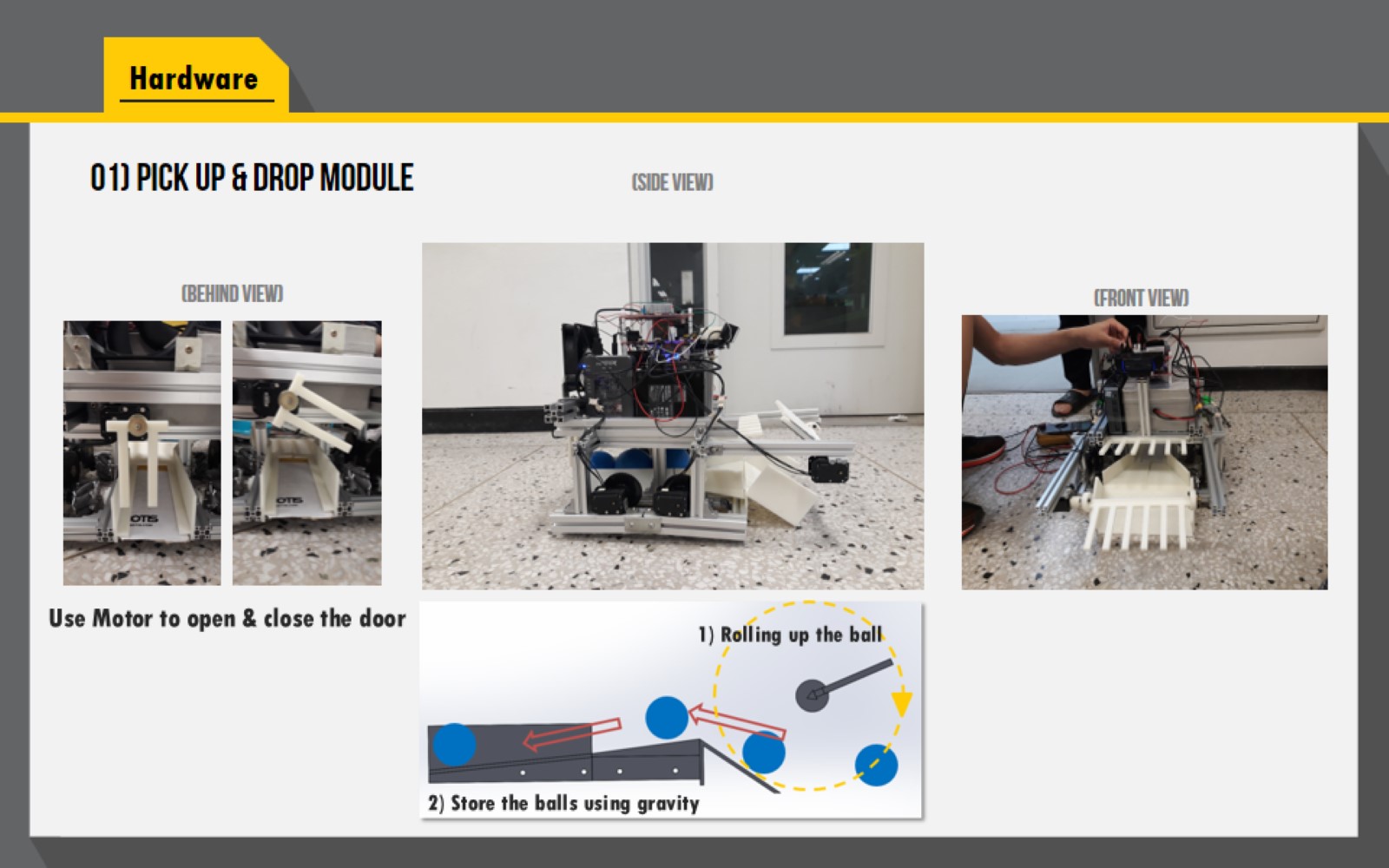
* 1. **Pick up module  
     **

Figure 2 pick up & drop module

다음은 공을 주워 담고, 내려놓는 모듈의 앞, 옆, 뒤에서 본 모습이다. 공은 차가 앞으로 이동하면서 앞 쪽에 있는 롤러에 의해서 앞쪽 경사면을 타고 들어올려진다. 경사면을 지나고 난 이후에는 중력에 의해 뒤쪽으로 굴러갈 수 있도록 경사로 형태로 저장소를 설계하였다.  
**1) Roller design**

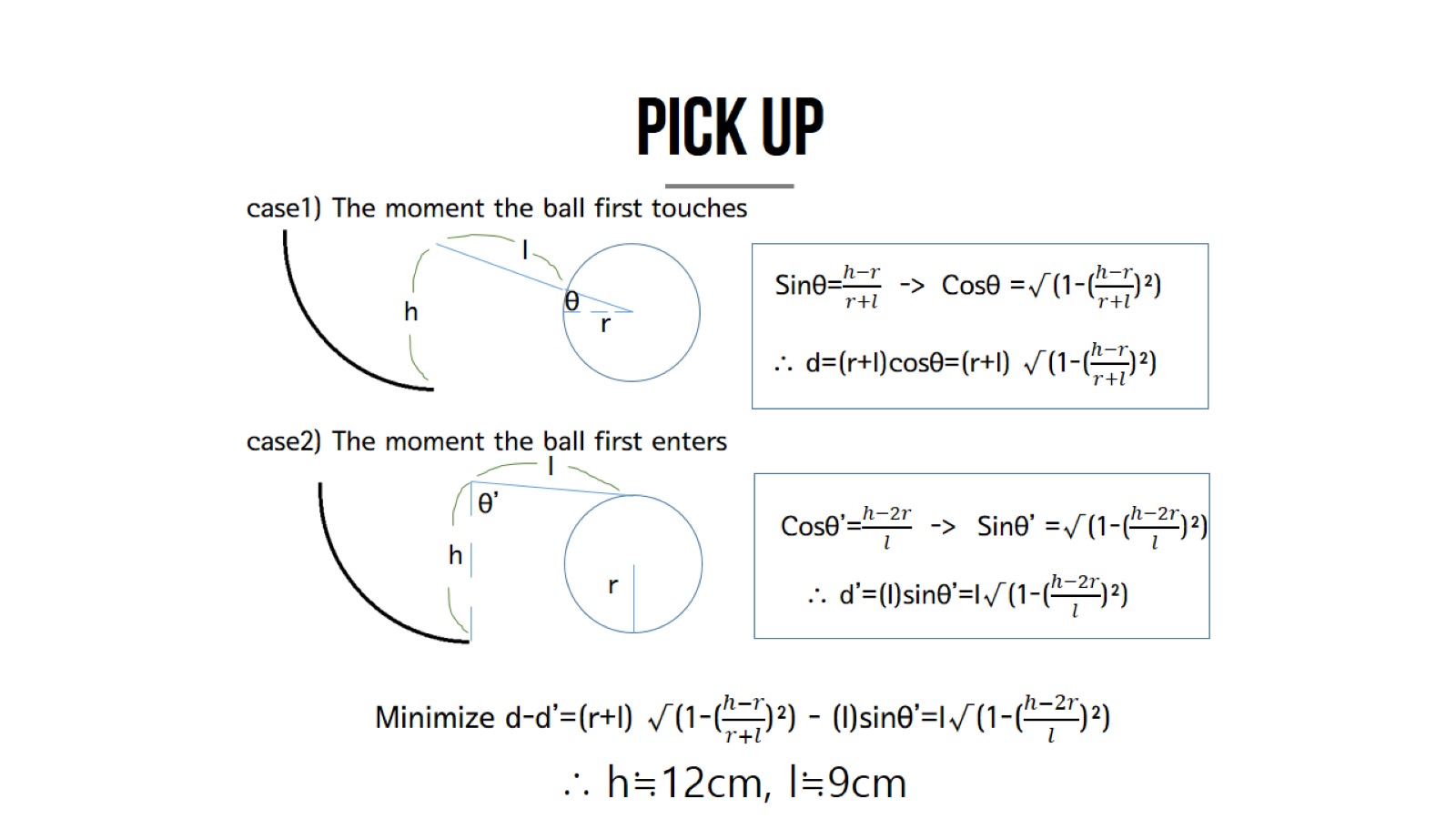


Figure 3 optimal dimension for roller

Figure 3은 공을 잡을 확률이 가장 높은 roller의 길이와 roller 중심 높이의 위치를 계산한 것이다. 그러기 위해서는 공을 놓치는 공의 위치 범위를 최소로 만들어야 한다. 그 결과 roller의 길이는 9cm, 중심의 높이는 12cm로 설정되었다.  
또한 roller에 의해 공이 다시 빠져나갈 수 없도록 위쪽에 가림 막을 설치하였다.

**2) drop module**

저장고가 뒤쪽으로 경사져 있기 때문에 뒤에 아무런 장애물이 없다면 공이 자연스럽게 떨어지도록 되어있다. 따라서 공을 주울 때는 저장소의 뒤쪽을 막아 두었다가 공을 떨어뜨릴 때에만 막아둔 것을 치우도록 했다. 다른 전원을 쓰지 않는 방법도 존재했지만 설계의 복잡도가 올라간다. 따라서 모터를 이용해 뒤의 장애물을 회전시켜 떨어뜨리도록 했다.

**3) rake shape**

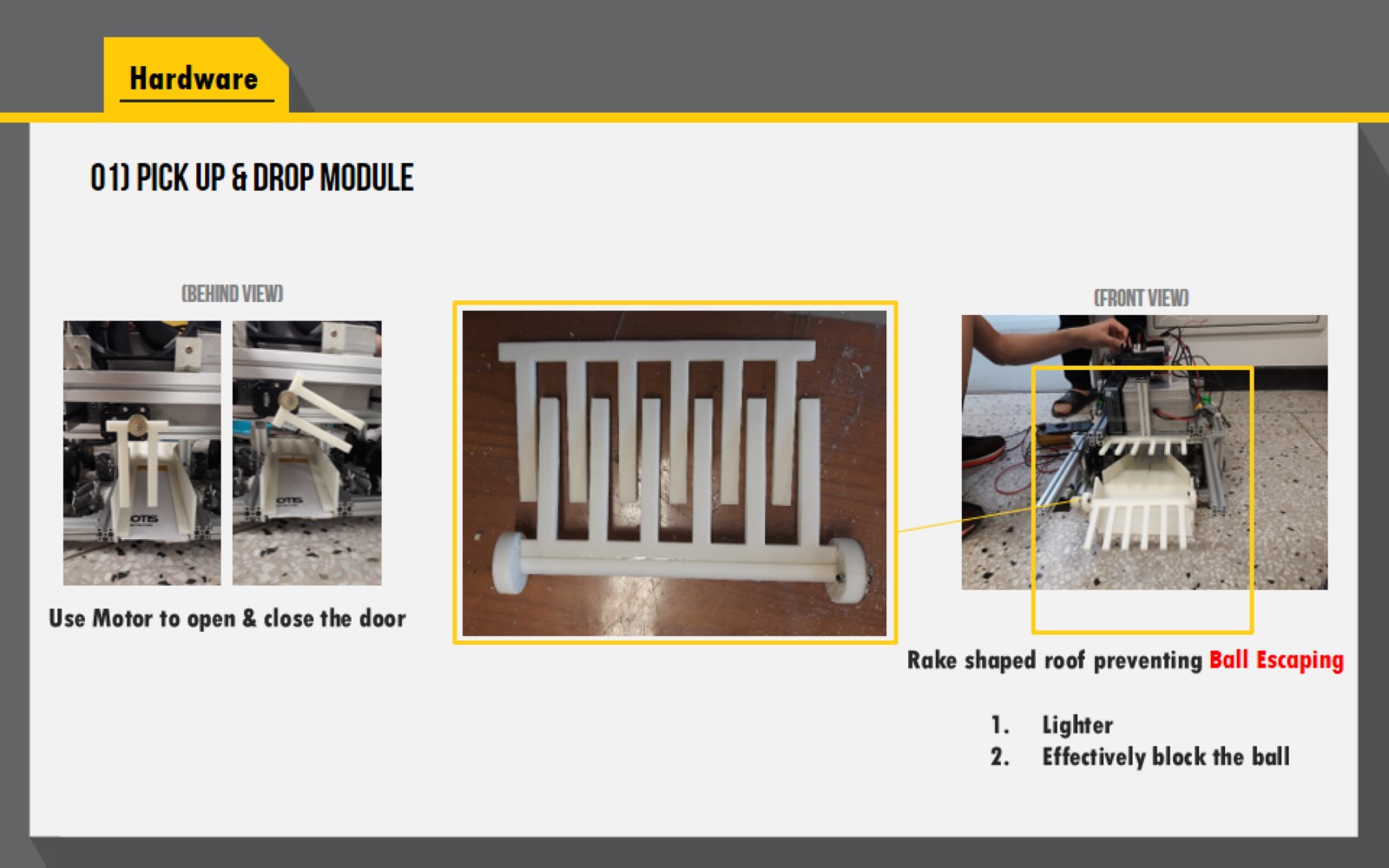


Figure 4 rake shaped

가림 막의 설치가 용이하고, 무게를 줄여 에너지 효율을 높이기 위해서 TRIZ 공간의 분리를 이용해 위와 같은 빗 모양의 구조를 사용하였다.

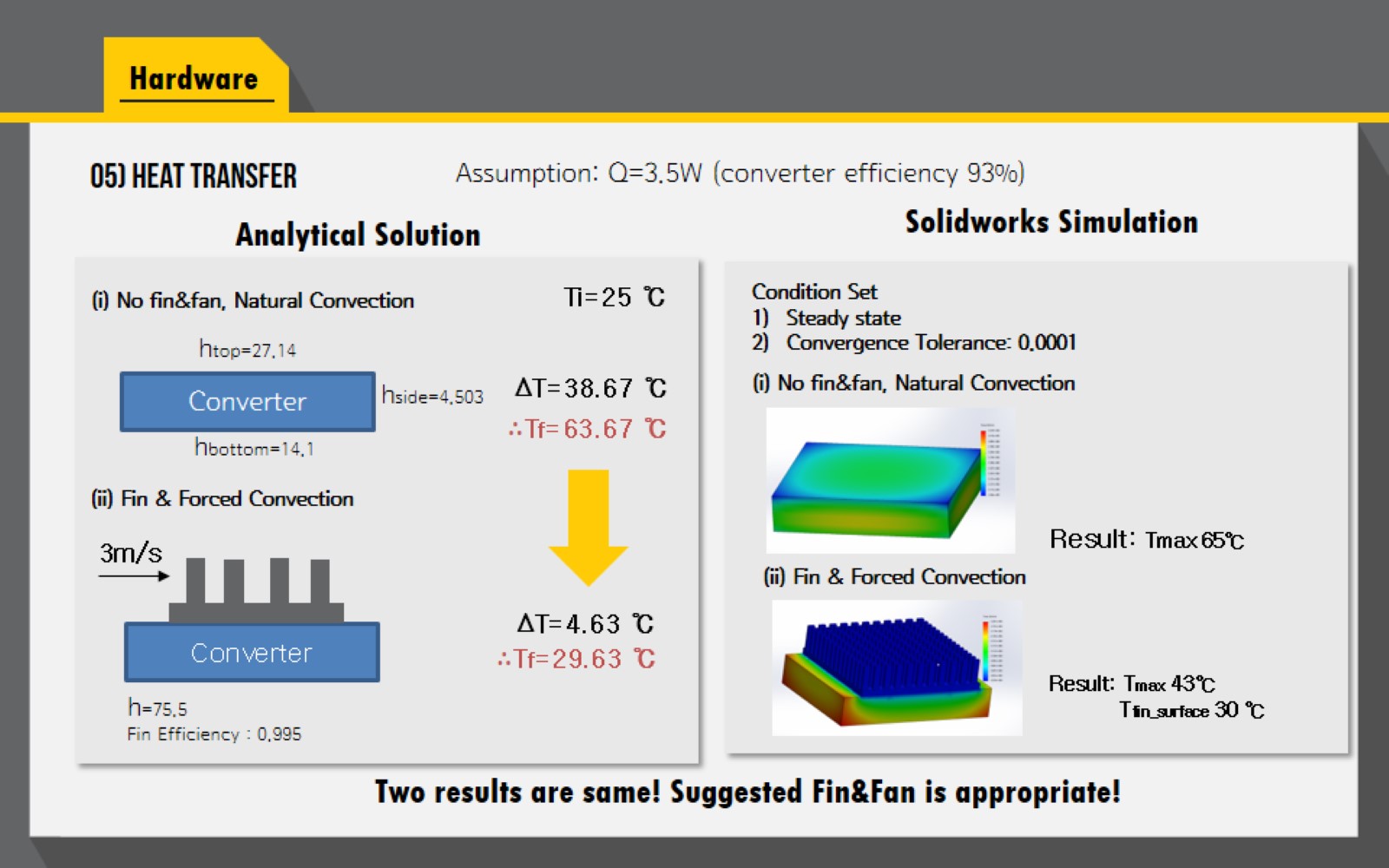
* 1. **Fan & Fin**차체의 최고 온도가 기준 온도인 70°C를 넘지 않았지만 가장 많은 에너지가 사용되기에 온도가 가장 높아졌던 모터 컨버터 위주로 냉각을 위한 설계를 진행하였다. Conduction은 냉매를 이용해야해서 설계의 복잡성이 커지고, radiation은 건물 내벽의 온도가 차체의 절대온도와의 차이가 크지 않기 때문에 영향이 적을 것으로 판단되어 fin과 fan을 이용한 forced convection을 사용하였다. ****

Figure 5 Heat analysis

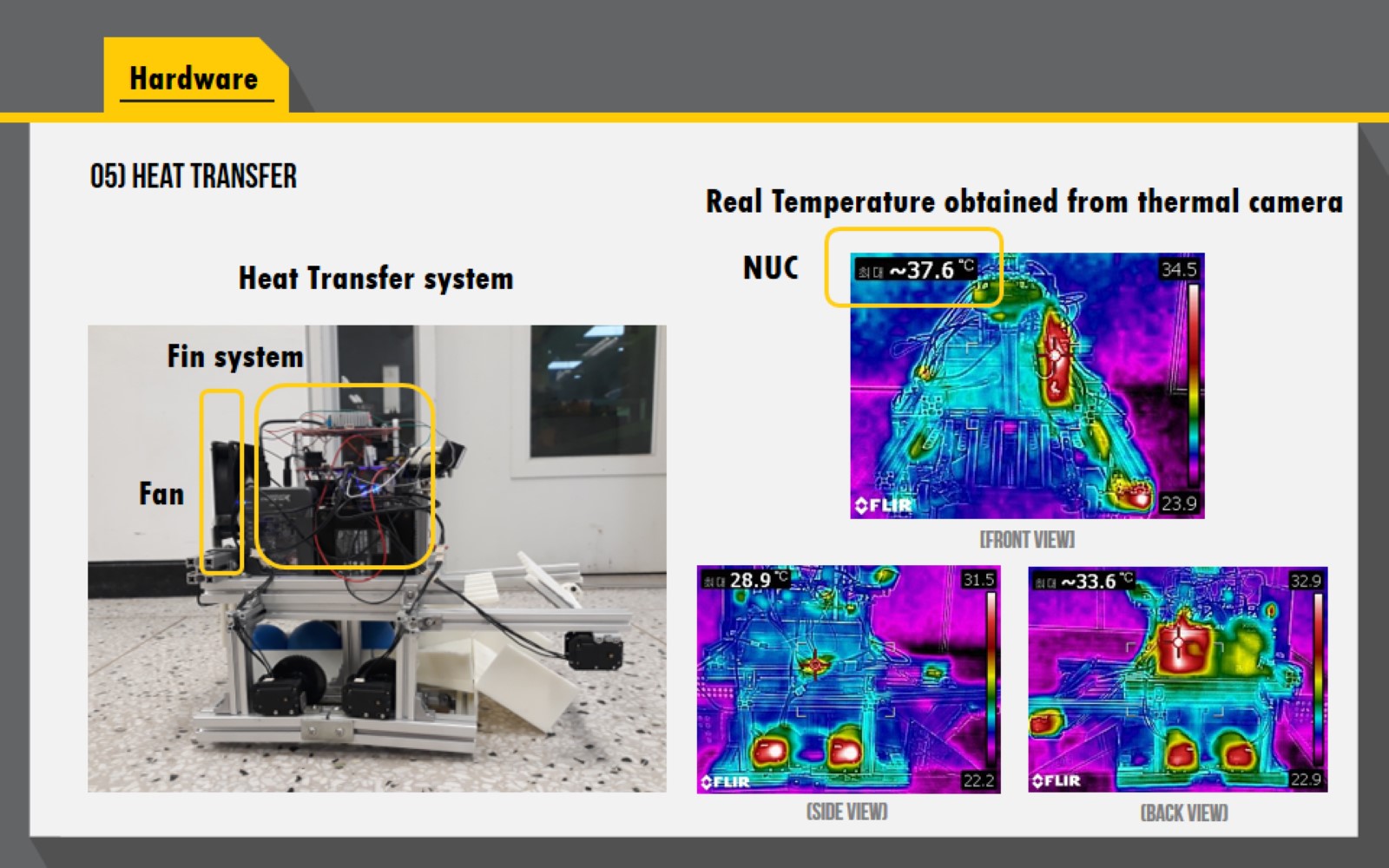
온도를 20°C이상 냉각시킬 수 있다는 것을 확인하였다.  
****

Figure 6 Temperature measured  
또한 실제 온도 측정 결과 또한 이론적인 분석을 뒷받침했다.

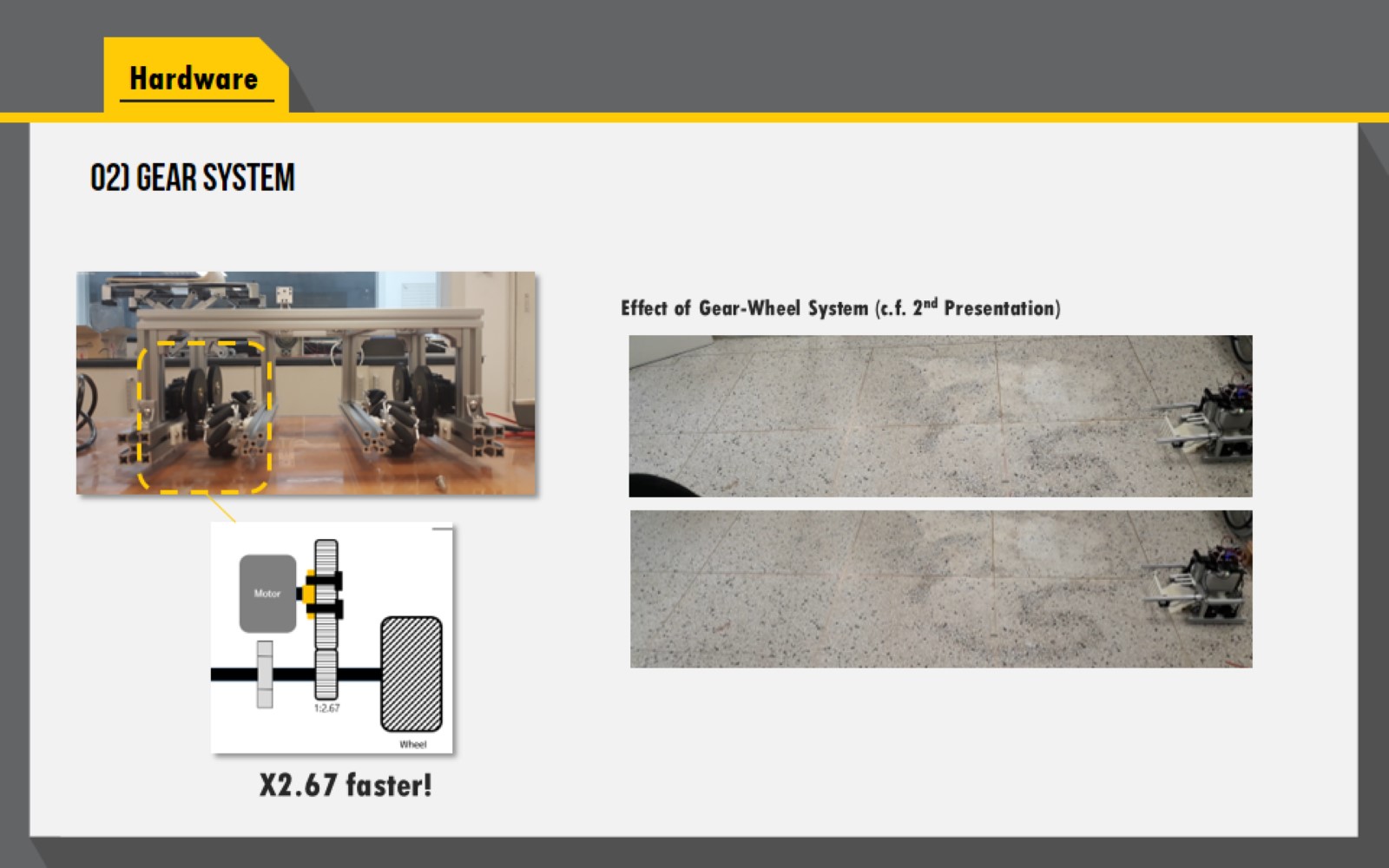
* 1. **Gear-Wheel system  
     **

Figure 7 Gear-Wheel system

기존에 주어진 모터 MX-28AT는 최대 55rpm으로 매우 느렸다. 따라서 기어를 사용하여 속도를 높였다. 기어는 바퀴의 크기 등을 고려하여 20:80 즉 기어 비 약 1:2.67로 설정했다. 결과적으로 약 2.6배 정도 속도가 빨라진 것을 확인할 수 있었다.  
또한 기어를 설계하면서 바퀴를 두 축으로 고정하면서 바퀴의 진동으로 인해 생기는 문제를 줄였다.

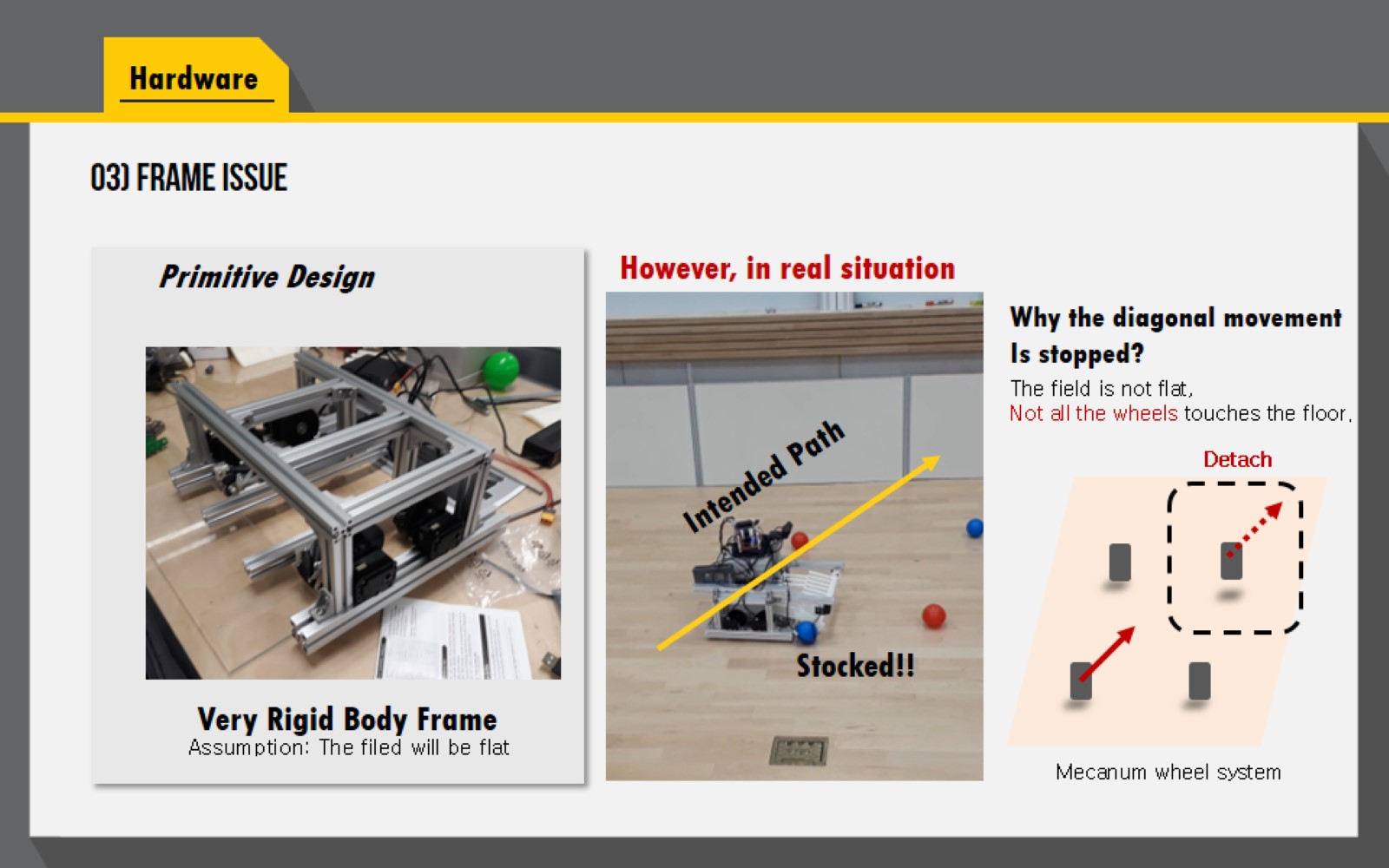
* 1. **차체  
     1) rigid body problem  
     **

Figure 8 rigid body problem

차체를 알루미늄 프로파일로 제작하고 바퀴를 두 축 고정하는 설계는 차체를 rigid하게 만들었다. 따라서 지면의 형태라는 noise로 인해 모든 바퀴가 땅에 고정되어 있지 않는 상황이 발생하였고, 이는 대각선 방향 진행에 큰 문제점으로 작용되었다. 따라서 차체에 자유도를 줄 수 있는 방법을 생각했었고, TRIZ기법을 이용하여 차체를 분리하여 회전 자유도를 부여할 수 있었다.  


Figure 9 Frame segmentation

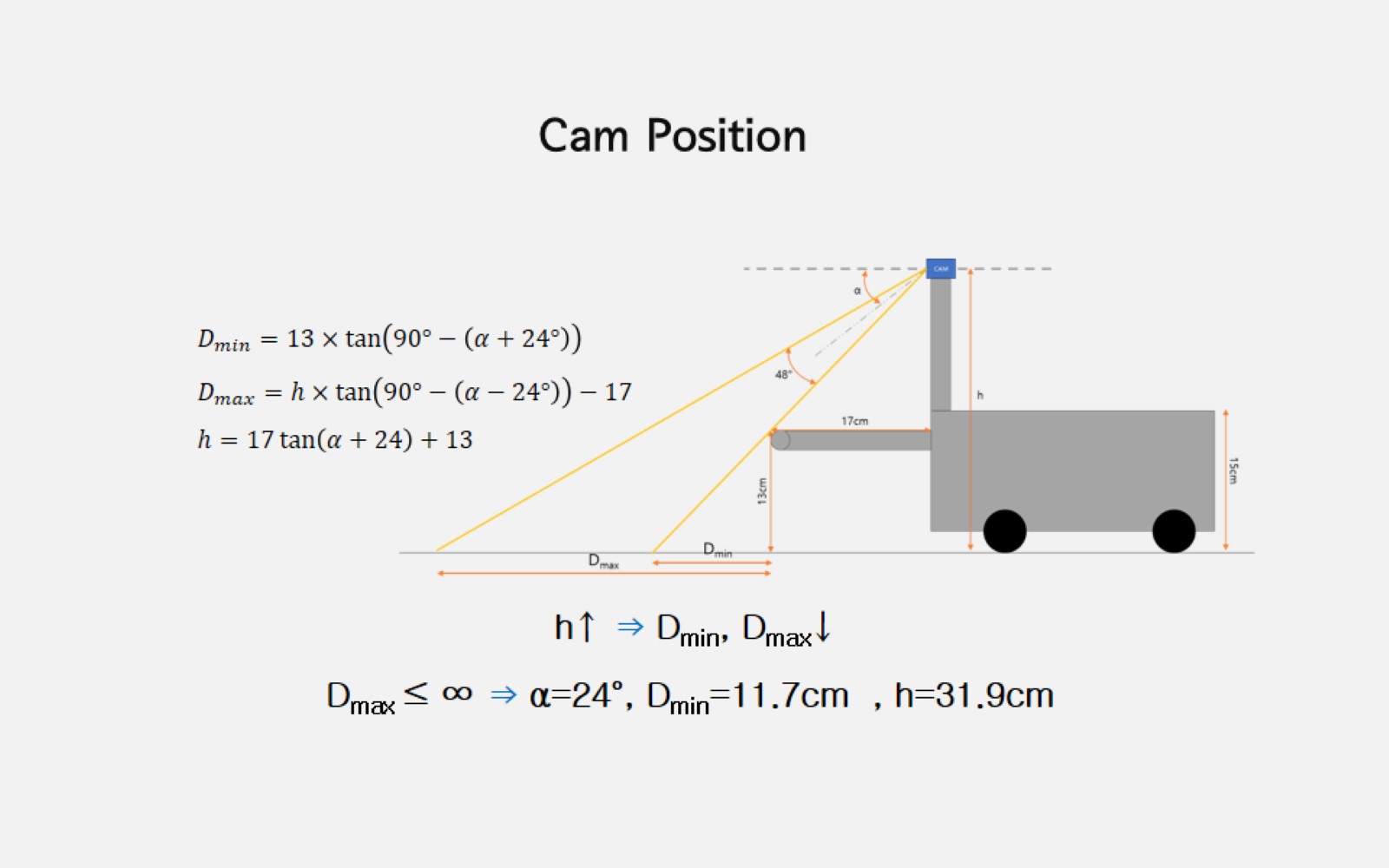
2)webcam위치 정하기  
최대한 가까운 곳까지 보면서 멀리 있는 공 또한 인식할 수 있도록 웹캠의 위치를 계산하였다. 대각선 시야각(FOV)이 76°라는 사실을 이용해 가로(66°), 세로(48°)의 시야 각을 추정해 계산했다.  


Figure 10 Webcam position

**2. Software Design**

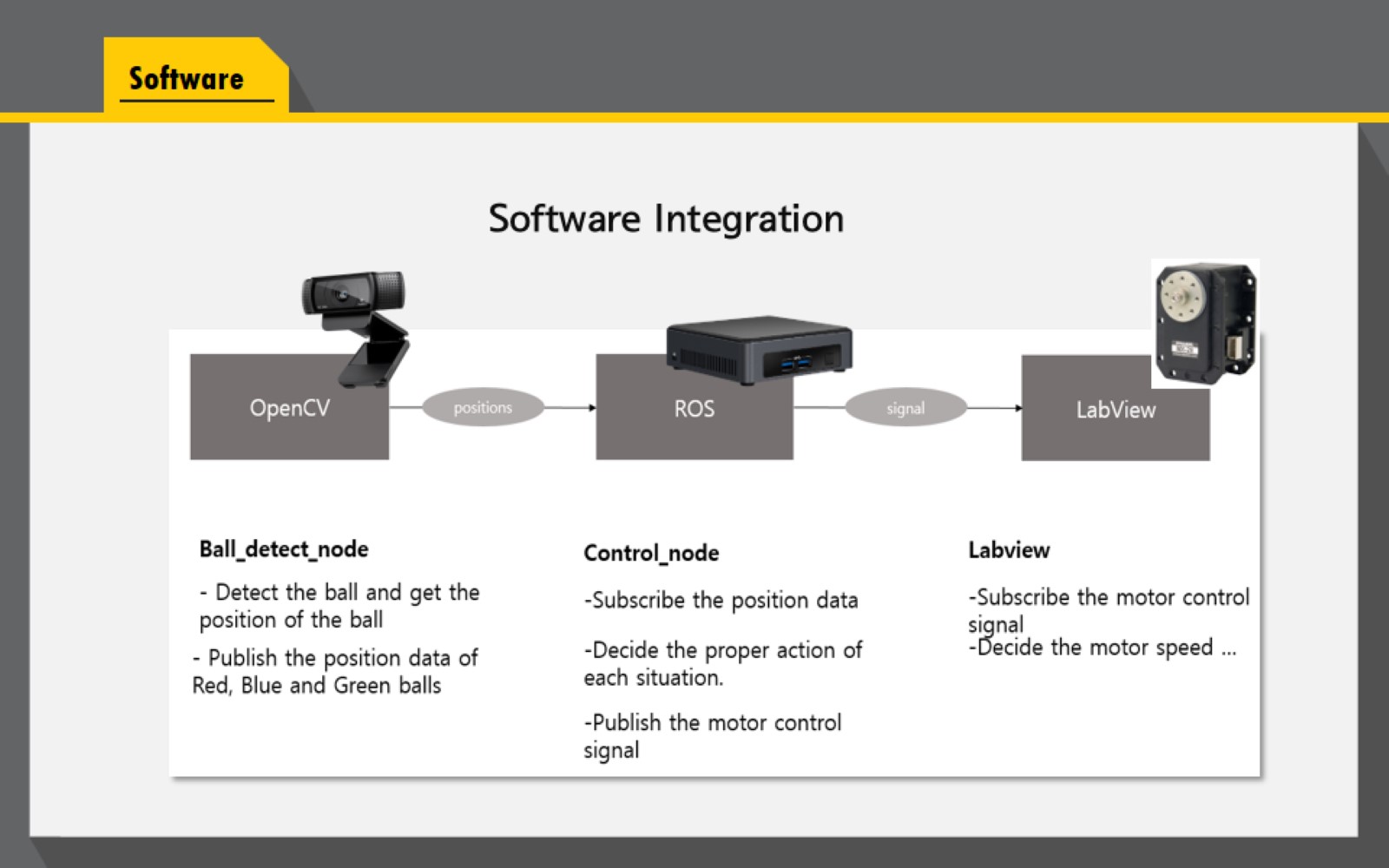
****

Figure 11 Software Integration

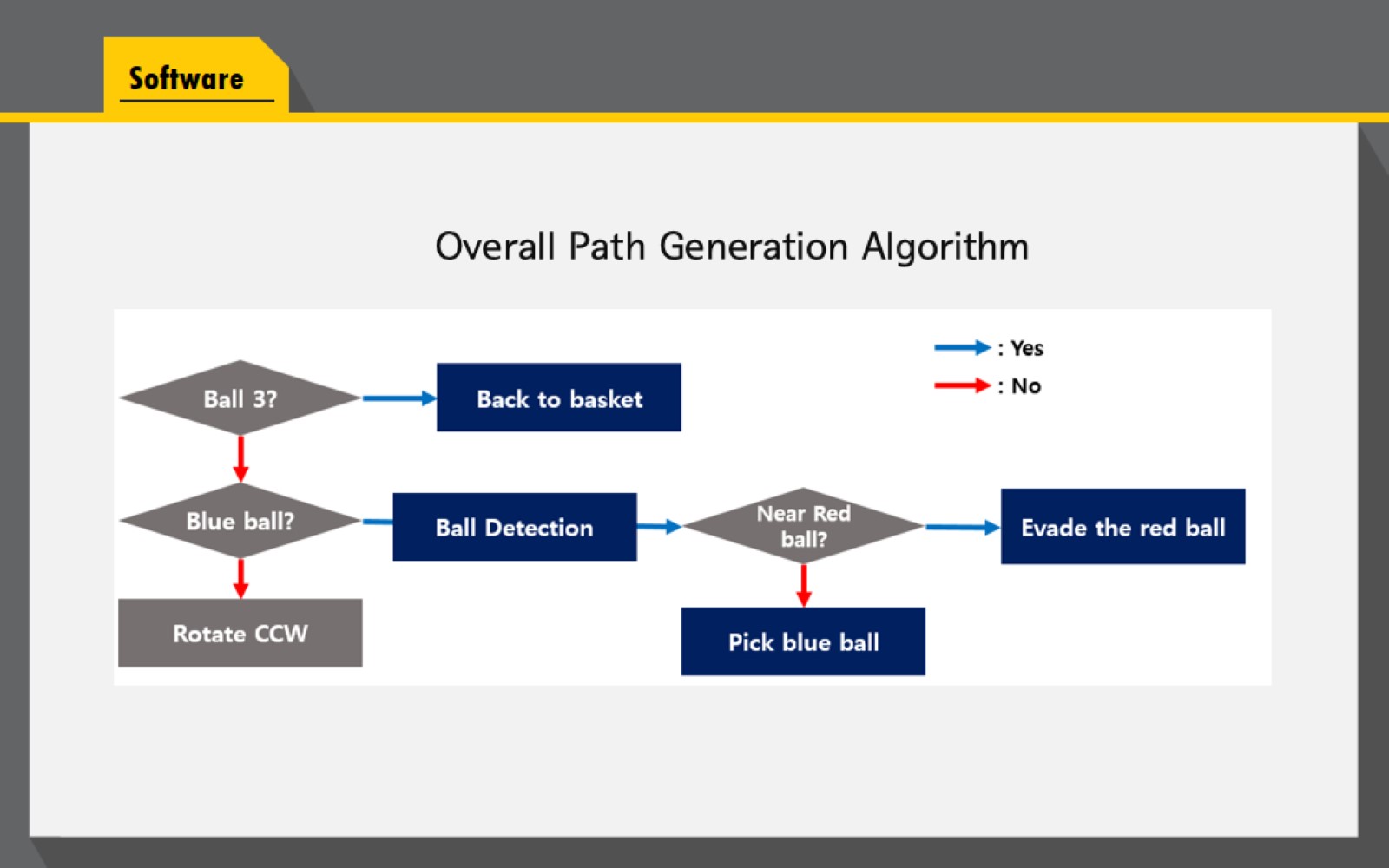
1. Overall Algorithm  
   

Figure 12 Overall Algorithm

전체 알고리즘은 다음과 같다. 공을 인식하고, 공에 접근하고, 파란 공이 가까워지면 차체와 파란 공 사이에 있는 빨간 공을 피하고, 공을 줍고, 공을 세번 다 주우면 바구니로 돌아간다.

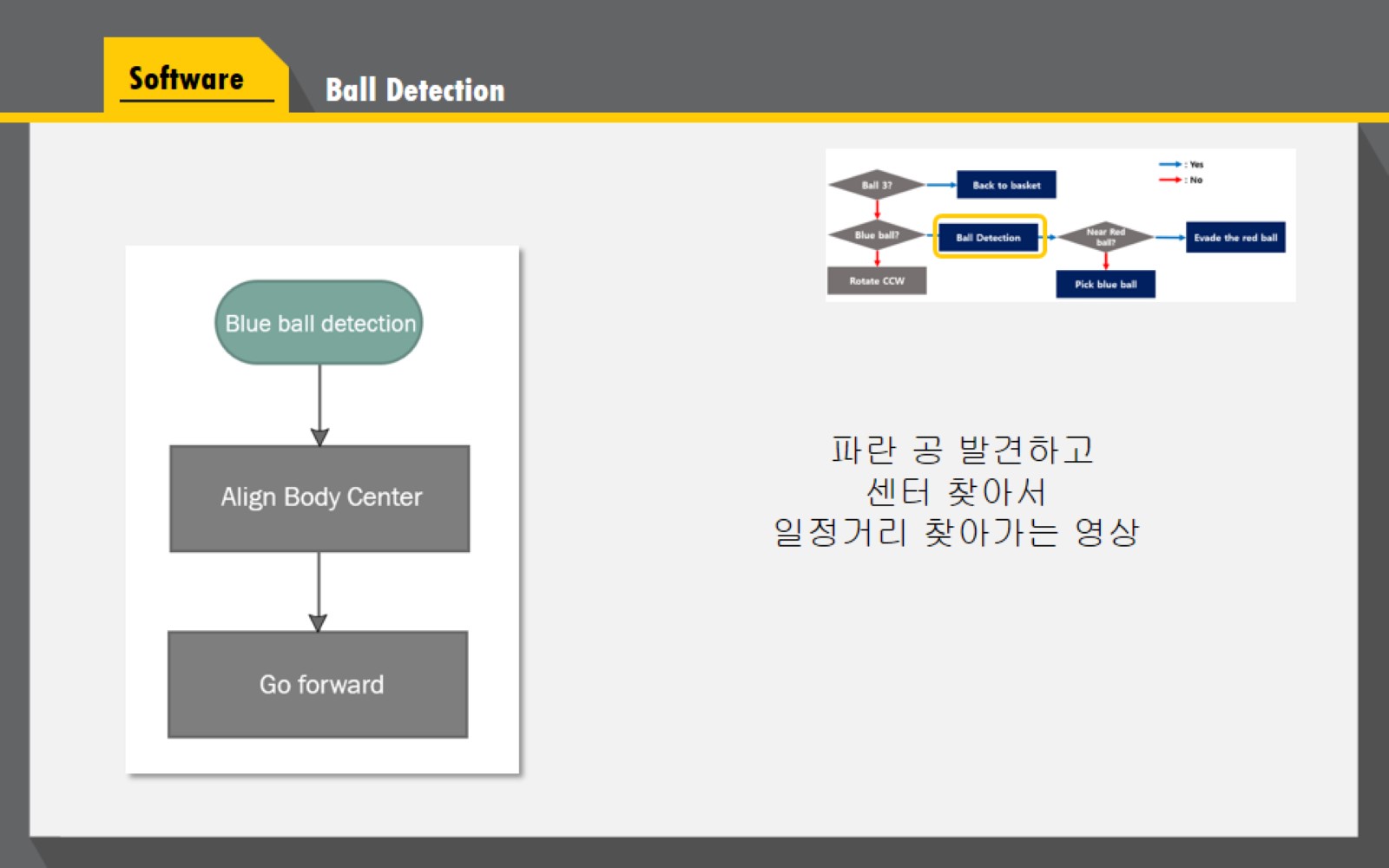
1. Ball detection  
   

Figure 13 Blue ball detection

파란 공을 찾으면 파란 공이 Mobile platform의 중앙에 오도록 정렬한다. 이 때 feedback control을 사용하여 disturbance가 있더라도 정확하게 공이 mobile platform의 중앙에 올 수 있도록 제어할 수 있다.

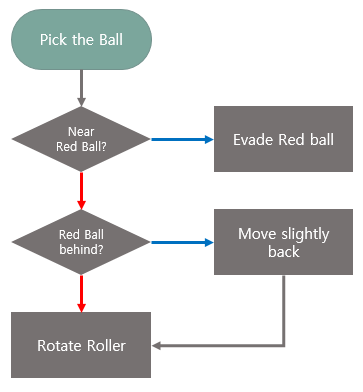
1. Pick the ball  
   

Figure 14 Pick the ball

파란 공 보다 가까이 있는 빨간 공이 있으면 대각선 움직임을 통해 빨간 공이 카메라의 중심으로부터 일정 이상 떨어질 때까지 Mobile platform을 이동시킨다. 그리고 파란 공을 줍기 위한 pick 동작에 들어가는데, 빨간 공이 파란 공의 뒤에 너무 가깝게 배치되어 있으면 약간 뒤로 움직여 빨간 공이 시야에 안보이는 상황을 방지한다. Pick이 1번 실행 될 때 마다 파란 공을 완벽하게 주울 수 있다고 가정하고, pick을 총 3번 실행할 때까지 공을 찾는 과정을 반복한다.

1. Back to the basket  
   

Figure 15 Back to the Basket

Pick이 3번 실행되면 바구니로 돌아가는 코드를 실행한다. 두 개의 초록색 공의 가운데를 target으로 잡고 바구니에 가까이 온 다음 대각선 움직임을 통해 두 녹색 공의 수직이등분선 상에 mobile platform을 위치시킨다. 이후 두 공의 중심과 mobile platform의 중심을 맞추고 feedback control을 통해 mobile platform이 정확하게 바구니에서 일정 거리 떨어지게 직진한다. 이후 두 개의 초록 공의 z좌표를 동일하게 맞추도록 회전 한 후 180도 회전한다. 그리고 공을 떨어뜨려 마무리한다.

**3. Process**

1. Monthly Plan  
   3월은 각자 파트를 공부하고 익히는데 중점을 두고 dynamixel이 도착한 4월, 5월에 본격적으로 프로젝트를 계획하고 진행하였다.  
   

Figure 16 Plan in April

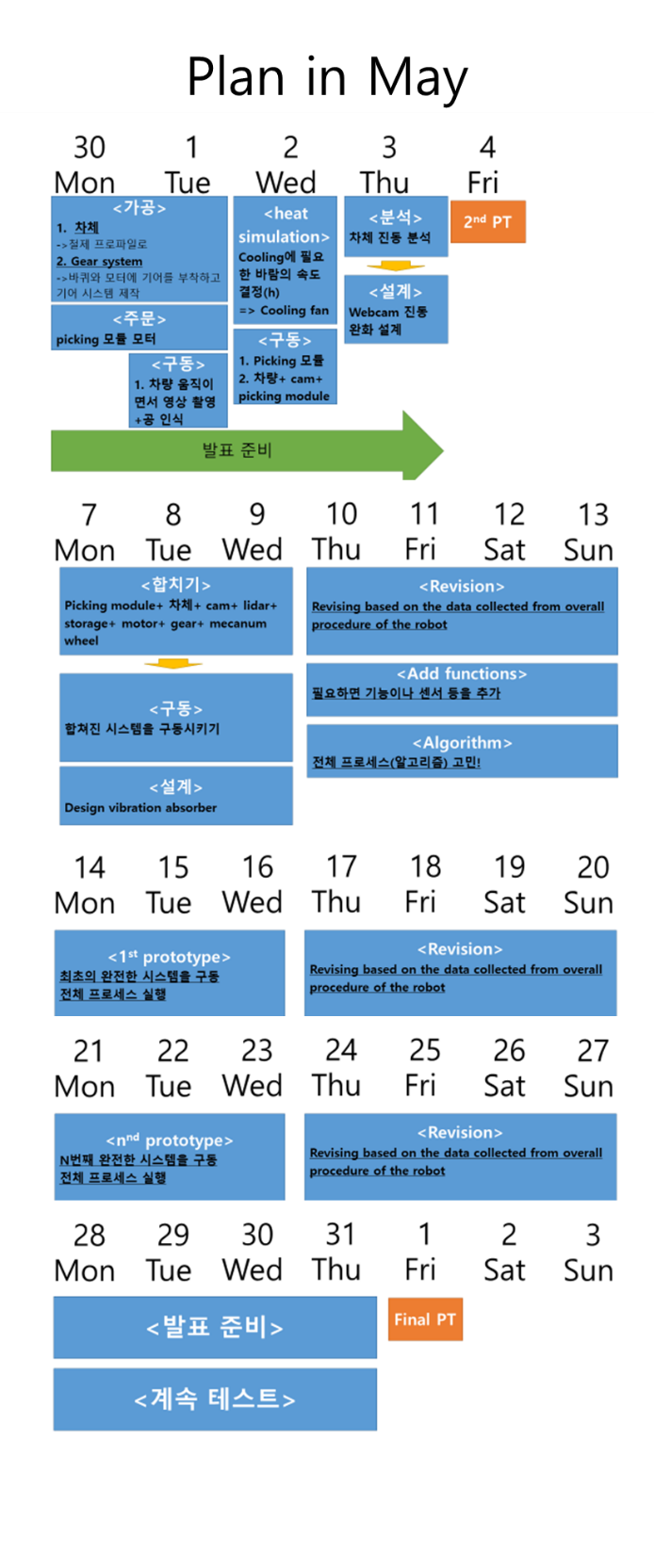


Figure 17 Plan in May

**3. Progress Report**

3월 19일 (월)

정기적으로 잡혀있는 박용화 교수님과의 미팅을 진행, 발표를 어떤 식으로 진행할 것인지에 대한 조언을 얻음

박용화 교수님께서 아이디어가 가장 중요할 것 같다는 내용의 조언을 해 주심. 이외에도 발표에 포함되면 좋을 것 같은 내용들을 몇가지 조언해 주심.

교수님과 미팅을 끝내고, 공을 집고 나서 확인하는 것을 컬러 센서로 확인하자는 아이디어와 3D 프린터로 대강 뽑아서 모델링을 빨리 해 보자는 의견을 냄.

Introduction of ROS 수업에 참여.

3월 20일 (화)

조별 미팅에서 전체적인 구조 설계. 조별 미팅이 끝난 후 ROS 조원과 모여서 assignment 1을 마무리함. ROS를 사용하여 node 생성, 한 노드가 여러 역할을 하도록 하는 방법, topic 통신, service 통신, roslaunch 함수를 사용하여 여러 패키지를 동시에 실행하는 방법 등을 익히고 활용할 수 있게 됨.

3월 21일 (수)

Heat Transfer 관련 공지 및 Heat Transfer 방법을 소개하는 수업에 참여. 조교님께서 수업 해 주신 내용 및 조언을 팀원들과 공유.

3월 23일 (금)

ROS code review session에 참여하여 조교님께서 나누어 주신 코드를 실행 및 전체적으로 ROS조가 해야 할 일을 명확하게 확인함.

박용화 교수님과 미팅을 가지고 발표에 관련된 조언을 한 번 더 들음. 월요일에 한 것과 비슷한 맥락으로 아이디어가 중요하다고 강조하심.

3월 26일 (월)

pick up 부분의 아이디어가 변화하였기 때문에 교수님과의 정기 미팅에서 이 부분의 아이디어에 대한 조언을 들음. 공을 줍는 데 pick up 부위와 mobile platform의 높이가 높아지며 불안정해 지는 상황이 발생하지는 않는지에 대한 analytic equation을 해결하여 화요일 조별 미팅에서 발표하기로 함.

3월 27일 (화)

조별 미팅에서 brush가 공을 잡기 위한 조건과 안정성의 조건에 대하여 analytic하게 푼 것을 설명해 줌. 조원들과 또다른 parameter가 무엇이 있을지에 대한 것을 토론.

ROS파트 조원과 함께 현재까지 우리가 어떤 부분까지 할 수 있는지, 정리하여 발표자에게 전달.

3월 28일 (수)

NUC에 우분투와 ROS를 설치하고 기본적인 통신이 가능한 상태로 설정함. 노트북과 무선 통신을 어떤 방식으로 할 수 있을지 시도해 봄.

ROS code review session에서 배웠던 라이다 실행, rviz로 라이다의 데이터를 시각적으로 보여주는 등을 직접 실행해 봄.

교수님과 미팅을 통해 발표 ppt의 구성, 수정사항, 추가했으면 좋을 것들 등에 대한 조언을 얻음.

교수님께서 조언해 주신 내용을 바탕으로 역할을 배분하여 ppt에 추가하기로 함. 앞서 했었던 analytic한 해석 부분을 맡아 이것을 그림과 수식을 통해 시각적으로 표현함. 발표자가 발표할 때 도움이 되도록 간단한 설명을 덧붙임.

3월 29일 (목)

교수님과 발표 전 마지막 미팅. Function decomposition을 포함시키는 방향을 고민해 보라고 하심. ppt의 구성에 대한 조언을 몇 가지 해 주심.

어제 내가 정리했던 brush가 공을 잡기 위한 analytic equation, 안정성에 대한 analytic equation을 영어로 간략하게 정리하여 발표자에게 전해 줌.

3월 30일 (금)

1차 발표 이후, 다음주의 계획 및 다음달의 계획에 대하여 간단하게 이야기를 나눔. 중간고사가 끼어있어 빡빡하게 진행해야 되겠다고 합의.

4월 2일 (월)

박용화 교수님과의 미팅에서 30일 세웠던 계획을 말씀드림.

4월 3일 (화)

크게 3개 조로 나뉘어 각자 해야 할 일을 맡아서 진행. Picking module dimension 설정 팀, 기어 비 설정 팀, 컨버터 스펙 설정, path generation 방식 설계, 모터의 토크를 결정 팀으로 나누고, 3번째 업무를 맡음.

NUC, MyRio, wheel의 motor에 현재 사용되고 있는 AC-DC 컨버터의 output 스펙을 조사함. 사용될 DC-DC 컨버터가 어떤 조건을 만족해야 하는지 조사함.

Path generation의 경우, 우리가 직접 프로그램을 작성하는 것 보다는 적당한 오픈소스를 가져와서 사용하는 것이 더 좋을 것 같다는 의견을 나눔.

Picking model에 사용되는 토크가 picking 동작을 위해 사용되는 토크가 어느 정도일지, 각속도가 어느 정도일지 analytic하게 분석하여 motor의 토크가 최대 0.1Nm정도, 각속도가 5rad/s 이상이면 충분하다는 결론을 내림.

4월 5일 (목)

프로토타입 제작 팀과 구입 제품 검색 및 검토 팀으로 나누어 구입 제품 검색 및 검토 역할을 맡음.

Picking module에 사용될 모터를 최종 결정. NUC에 사용될 컨버터, MyRio에 사용될 컨버터, wheel actuator에 사용될 컨버터, Picking module에 사용될 컨버터를 모두 결정함.

기어 구입처도 찾았으나, 기어의 경우 낱개 판매하는 지도 확실치 않고 가격도 정확하게 책정되지 않아 여상은 선생님께 문의 드린 후 최종 결정하기로 함.

조원 전체가 모여 추가적으로 전원케이블이 필요하다는 의견에 따라 전원케이블도 구입하기로 결정함.

4월 6일 (금)

여상은 선생님과 연락하여 기어 구입에 대한 조언을 들음. 기어는 내가 찾은 사이트에서는 배송이 늦어질 수 있다고 하여 같은 제원으로 대전에 주문제작 기어에 견적을 내 주시겠다고 해 주심. 추가로 기어에 연결할 베어링, 축 등을 설계하기 위해 직접 한번 찾아오라고 하셔서 조원들과 월요일에 찾아 뵙기로 함.

개인적으로 리눅스에서 g++ interpreter를 활용하는 방법을 학습함.

4월 9일 (월)

여상은 선생님과 만나 기어의 설계에 대한 조언을 얻음. 미팅 후 기어 설계를 확정함. 기어를 고정하는 방법, 다이나믹 셀과 축을 연결하는 방법 등 많은 방법들을 고안해 보고 최종적으로 기어 설계를 확정함. 대화동에 나가 물품들을 구입하기로 결정함.

4월 11일 (수)

대화동에서 베어링, 열처리가 되지 않은 축, 커넥터 등을 구입함. 우리가 기어를 설계할 때 필요한 부품들은 베어링을 제외하고는 주문제작해야 한다는 것을 확인함. NUC에 사용될 컨버터의 전압이 너무 낮다는 의견이 나와 컨버터의 스펙을 다시 조사해 보기로 함.

4월 13일 (금)

기어비와 컨버터를 결정하기 위해 모임. 여상은 선생님이 제공해 주신 카탈로그에서 기어 이빨의 수를 30, 60, 80으로 하여 주문하기로 최종 결정함. 컨버터의 경우, NUC에 연결될 컨버터만 교체하기로 결정하고 컨버터의 주문을 최종적으로 완료함.

3D 프린터로 뽑은 pick up module 부분의 부품들도 합쳐서 제작해 봄.

4월 15 ~ 20일(시험 주)

xbox\_ctrl node가 어떻게 구동 되는지 확인. 노트북 내에서 서버를 열고 통신이 되는지 확인해 봄. 주어진 ROS textbook을 읽고 SLAM이 어떤 개념인지 학습함. Transformation matrix에 관련된 예제들을 실습해 보고 frame의 transformation에 대해 이해함. 기어 관련하여 주문을 여상은 선생님과 연락하여 마무리함.

4월 22일 (일)

ROS에서 xbox\_ctrl node를 실행하고 myRIO와 통신하기 위한 시도를 해 봄. ROS에서는 올바를 신호를 보내고 있으나, myRIO에 데이터가 도착하지 않는 것을 확인함. LabView 코드를 수정하기 위해 LabView 파트 조원들과 토론을 지속했지만 해결하지 못함.

4월 24일 (화)

다시 LabView 파트 조원들과 만나 통신을 하기 위한 토론을 지속함. LabView에서 TCP 통신에서 받는 데이터를 어떤 형식으로 받는 지 이해하고, LabView 코드를 수정함. 데이터를 자르는데 사용되는 byte와 case가 반복되는 시간을 수정하여 ROS에서 Xbox의 데이터를 전송하는 것을 myRIO에서 실시간으로 받아올 수 있도록 코드를 완성함.

4월 26일 (목)

openCV 파트 조원과 만나 ball\_detection node와 openCV 파트 조원이 작성한 코드를 합치는 작업을 함. 그리고 금요일에 있을 ROS 세션에서 발표를 하기 위한 process diagram을 작성함. openCV 코드와 합치고 난 후, ball의 위치는 잘 받아오는 것을 확인하였으나 이미지 출력이 되지 않는다는 사실을 확인하고 해결하기 위해 노력하였으나 해결하지 못함.

Xbox를 통해 myRIO와 통신할 때 myRIO와 노트북 사이의 와이파이 연결이 매우 불안정하다는 사실을 파악함. myRIO의 기기의 문제일 것이라 추측.

4월 27일 (금)

ROS세션에서 현재까지의 진행 사항을 발표함. 현재까지 진행된 것까지의 process 정리는 좋으나 후반부의 process를 조금 더 정리해야 할 것 같다는 코멘트를 받음.

조교님의 도움을 받아 openCV 코드를 수정하여 이미지 출력이 되도록 코드를 수정하였으나, 이번에는 반응속도가 매우 느린 오류를 발견하고 이 부분을 해결함.

Mobile platform에 배터리, myRIO를 싣고 노트북에서 와이파이를 연결하여 Xbox controller를 통해 mobile platform의 움직임을 제어할 수 있게 됨.

4월 29일 (일)

전체적인 Process 수정. 현재까지 진행된 사항과 앞으로의 계획을 PPT로 제작함.

4월 29일 (일)

교수님과의 미팅, 2차 발표 때 어떤 내용을 중점적으로 발표할 것인지 교수님께 말씀드린 뒤 analytic한 해석이 조금 더 필요할 것 같다는 피드백을 받음.

5월 1일 (화)

공을 detect하고 자동으로 공을 주으러 가는 알고리즘을 확정함. 공의 개수, 파란 공만 detect 되는 경우, 파란 공과 빨간 공이 동시에 detect 되는 경우, 파란 공이 detect 되지 않는 경우 등으로 경우를 나누어 움직임을 결정하도록 코드를 작성함. NUC에 모든 코드를 얹고 myRIO, NUC, 창시구실의 컴퓨터를 사용하여 Mobile Platform이 작성된 코드에 따라 움직이는지 확인해보려 했지만, 기어가 제대로 고정되지 않으며 차체가 불안정 해 짐. 다음날 솔리드웍스 파트 팀원이 차체 설계를 수정하면 다시 Mobile Platform이 잘 작동하는지 테스트 해 보기로 함.

5월 2일 (수)

하드웨어 부분의 설계가 안정적으로 완료되어 소프트웨어 부분을 시험해 볼 수 있게 됨. 다만 PMS의 납땜 부분의 문제와 선 연결 문제가 발생하며 해당 부분을 수정하고 나서 알고리즘을 실험해 본 결과 예상했던 것 보다 안정적으로 detecting을 해 낸다는 사실을 확인함. 파란 공을 pick up하는 파트에서 Mobile Platform의 움직임을 조금 더 안정적으로 하기 위한 수정이 필요할 것 같고, pick up motor를 구동시켜야 한다는 추가적인 과제가 남아있음.

또한 공의 y좌표를 출력하는 함수를 추가하여 카메라의 진동을 파악할 수 있도록 함. 또한 이것을 매트랩으로 plot하여 카메라의 진동을 해석해야 할 필요성이 있다고 주장함.

5월 3일 (목)

교수님과의 미팅 진행. 특별히 발표를 위해 요구하시는 사항은 없었음.

모터 구동 파트를 코드에 추가함. 먼저 구동 되도록 하는 것이 목적이기 때문에 픽업을 할 때 공이 카메라 시야 밖으로 나가기 때문에 공이 일정거리 이상 가까워지면 수동적으로 일정 거리를 움직이도록 코드를 작성함. 실제 여러가지 데모 사정들을 상정하고 실제로 다양한 상황들을 시험해 본 결과 정확도를 조금 더 상승시켜야 할 필요성이 있다고 생각됨.

파란 공을 Mobile Platform의 중간에 놓는 코드에 길이 조건 뿐만 아니라 각도 조건도 추가함. 발표에 사용하기 위한 여러가지 동영상들을 촬영함.

그리고 PMS에 대한 열 해석과 카메라에 대한 진동 해석을 한 후 발표자료에 추가함.

5월 4일 (금)

발표 후 다음 조별 미팅 시간을 협의함.

5월 8일 (화)

마지막 데모까지 해야할 일들을 정리하고 역할 분담을 함. ROS 파트 조원과 함께 파란 공이 3개가 들어왔다는 사실을 어떻게 판단 할 것인지, 만약 컬러센서를 사용한다면 어떤 센서를 사용할 것인지 조사하는 역할을 맡음. 또한 전체 알고리즘 수정, 라이다를 어떻게 활용할 수 있을지도 생각 해 보기로 함.

5월 11일 (금)

LabView 파트 조원과 ROS 파트 조원이 같이 모여 pick up 파트의 모터, backdoor 모터, 공 3개를 어떻게 판단할 것인지 등을 고찰함. Backdoor의 경우 현재 경사로의 마지막 부분이 목표 바구니보다 낮기 때문에 경사로의 높이를 높여야 할 필요성이 있음. Pick up 파트의 경우 모터가 문제가 있는 것이 아닌 경사로의 문제가 있는 것으로 판단되어 경사로를 곡선이 아닌 직선으로 경사를 조금 더 낮게 하고, backdoor 부분 때문에 높이가 높아짐에 따라 적당한 경사를 얻기 위해서는 pick up 모터가 달린 팔을 조금 더 길게 달아야 할 것 같다는 결론을 내리게 됨.

공을 3개 주웠다는 사실을 확인하는 것에 대해서는 압력센서, 컬러센서, 카메라 등의 의견이 나왔으나 하나의 사실로는 전부 확인할 수 없다는 결론을 내림. 따라서 ‘pick up 모터의 구동이 몇 번 발생하였는지’ 와 ‘컬러센서에 파란색이 감지되는가’ 를 동시에 판단하여 3개의 공을 잡았는가를 판단하기로 함.

또한 myRIO와 NUC에서 모두 신호가 잘 가고 있으나 모터가 구동되지 않는 PMS에 고질적인 문제점이 다시 발생함. 이를 해결하기 위해 선을 다시 확인하고 교체하는 작업을 수행함. 이후 Mobile Platform을 구동시켜 본 후 현재 pick up 모터의 속도가 적절하다는 결론을 내림.

5월 13일 (일)

조원들과 함께 하드웨어의 디자인을 바구니에 맞추어 다시 설계해야 한다는 결론을 내리고 어떻게 설계할지에 대한 이야기를 나눔. Backdoor의 디자인을 창의적으로 할 수 있는 방법들에 대해서도 이야기를 나누어 봄.

진동 해석에 대해 내일 교수님께 여쭤보기 위해 직진, 회전, 대각선 움직임에 대한 데이터를 추출하고 매트랩으로 plot 해봄. 교수님에게 우리가 사용하고 있는 방법이 해석할 때 적절한지 조언을 얻기로 결정함. 특히 우리가 얻을 수 있는 데이터가 5Hz 주파수의 discrete한 데이터인데, 이것을 사용하는 것이 과연 적절한 일인지, 어떤 방법으로 진동 해석을 할 수 있을지를 중점적으로 질문하기로 함.

5월 14일 (월)

교수님과의 미팅에서 한정적인 데이터를 사용하여 진동을 분석하는 방법에 대해 조언을 얻음. 지금의 sampling rate에서의 분석은 거의 의미가 없는 것이라 판단되어, 최대한의 sample을 얻을 수 있는 30Hz의 sampling rate를 사용하여 진동을 분석하기로 함.

5월 16일 (수)

교수님의 조언을 토대로 진동 분석을 다시 실행함. 추가로 열전달에 필요한 fin과 fan을 주문하고 후진할 때 바구니까지의 거리를 쉽게 알 수 있게 하기 위해 myRIO에 호환되는 초음파 센서를 찾아서 주문함.

5월 17일 (목)

공동강의실에서 첫 데모를 진행함. LabView를 실행하는 컴퓨터가 없었기 때문에 Mobile Platform을 이곳저곳 옮겨가면서 실제 상황에서 공을 얼마나 잘 인식하는지, 얼마나 먼 거리의 공도 인식할 수 있는지 확인하여 라이다를 사용하지 않아도 충분히 바구니의 초록색 공을 인식하고 돌아올 수 있을 것 같다는 결론을 내림. 들어오는 공을 확인하는 알고리즘을 센서를 사용하지 않고, pick 모드가 3번 작동하면 파란 공을 3개 주운 상황이라고 가정하고 basket으로 돌아오는 코드를 작성함. 바구니로 돌아오는 알고리즘으로 인식되는 초록색 공들의 좌표들의 평균값을 목표값으로 하여 움직이게 하자는 의견을 제시하고 이를 구현함.

5월 22일 (화)

공동강의실에서 처음으로 Mobile Platform을 작동시켜 봄. 대각선 움직임이 창시구실에 비해 매끄럽지 않다는 것을 확인함. 공동강의실의 바닥이 꽤 울퉁불퉁한 데 반해, 우리조의 Mobile Platform은 매우 rigid한 system으로 만들었기 때문에 바퀴가 떠서 움직이지 않는 상황이 발생한다는 것을 확인함.

5월 23일 (수)

교수님과 공학설계실에서 만남을 갖고, 대각선 움직임에 대한 조언을 얻음. Rigid한 system에 자유도를 조금 더 줘서 바퀴가 자중에 의해 울퉁불퉁한 바닥 위에 있더라도 접촉이 가능하도록 하는 것이 좋겠다는 조언을 얻음.

뒷문 모터를 ROS에서 주는 신호를 통해 움직여 보기로 하였으나, 작동하지 않음. 납땜, 선 등을 추가로 확인해야 할 것 같음.

5월 25일 (금)

데모에서 대각선 움직임이 알고리즘에 많이 포함 될 경우, Mobile Platform의 구조에 무리가 많이 간다는 사실을 확인함.

또한 기존의 가까운 공부터 집는 알고리즘을 사용할 경우 고질적인 문제인 다음 공을 발견할 때까지 시간이 매우 많이 걸리는 상황이 발생하는 것을 해결하기 위해, 왼쪽 혹은 오른쪽에서부터 공을 집기 시작하여 한쪽방향으로 회전을 하면서 순서대로 공을 집는 알고리즘이 더 좋을 것 같다는 의견을 모음.

기존의 알고리즘에서 왼쪽 방향에서부터 공을 수집하는 알고리즘으로 바꾸어 구현함. 기존의 알고리즘은 가장 가까운 파란색 공을 인식하는 것인데, 먼저 Mobile Platform을 왼쪽으로 보낸 후 가장 왼쪽에 있는 파란색 공을 가장 먼저 가지고 가는 알고리즘으로 변경함.

5월 26일 (토)

데모에서 왼쪽부터 공을 수집하는 알고리즘이 대각선 움직임이 매우 많이 발생하는 사실을 확인함. 기존의 방법과 시간을 비교했을 때 회전으로 손실되는 시간과 대각선 움직임에 의해 손실되는 시간 사이의 차이를 비교하여 더 나은 방법을 찾아야 할 것이라는 결론을 내림.

바구니로 돌아오는 알고리즘도 기존의 알고리즘이 바구니에 공을 담지 못하는 것 같아서 알고리즘을 수정함. 두 초록색 공의 수직 이등분선 위에 먼저 Mobile Platform을 보낸 후 정확히 바구니를 보도록 한 후 180˚ 회전시키는 코드를 작성함. 배터리 방전으로 코드의 완성을 미루기로 함.

5월 27일 (일)

배터리를 충전한 후 알고리즘을 수정하던 도중, 알고리즘에 문제가 없음에도 원하는 움직임을 구현하지 못하는 것을 발견함. 원인을 찾던 도중, 하나의 공을 여러 개의 공으로 인식하는 문제가 있는 것을 발견. OpenCV 파트 조원이 공의 인식 문제를 해결하고, Control node에서도 자체적으로 초록색 공에 대해 X축을 기준으로 가장 오른쪽에 있는 것과 가장 왼쪽에 있는 것 두 개만을 받아오도록 하는 코드를 작성함.

5월 28일 (월)

공동강의실에서 데모를 진행하고, 180도를 돌기 위해서 몇 초를 회전시켜야 하는가 등을 체크해 봄. 그리고 왼쪽부터 시작해서 공을 집어 오는 코드, 가장 가까운 공부터 시작해서 공을 집어 오는 코드, 오른쪽부터 시작해서 공을 집어 오는 코드 등을 모두 실험해 봄. 각각의 방법들 모두 최악의 상황이 발생할 것이라는 결론을 내림. 결국 가장 가까운 공부터 시작하여 공을 집어오는 코드로 데모를 진행하기로 결정함.

5월 29일, 30일, 31일 (화, 수, 목)

계속해서 Mobile Platform의 속도, Pick up motion을 할 때의 속도 등을 튜닝하는데 집중함. 원래 사용하고자 했던 pick up machine에 사용되는 모터가 고장나 다른 모터를 사용하게 되어 기존에 목표로 했던 스펙들을 맞추기 어렵게 되어 다른 조건들을 바꾸면서