

[두산로보틱스] 지능형 로보틱스 엔지니어

내가 그린 그림대로 도미노 (협동1)

F-1 조

3인조

한준모, 배재성, 김동호, 김예신(중도포기)

[멘토] 김민수, 강현우

K-Digital Training

목 차

- 01 프로젝트 개요
- 02 프로젝트 팀 구성 및 역할
- 03 프로젝트 수행 절차 및 방법
- 04 프로젝트 수행 경과
- 05 자체 평가 의견

01

K-Digital Training

프로젝트 개요

1

프로젝트 주제 및 선정 배경, 기획의도

시중에서 판매는 제품들은 pick&place와 같은 간단한 task만 예시 구현, 사람들이 쉽게 접근할 수 있으며, 조금더 여러 기능을 사용하며 복잡한 task를 구현하기 위한, 도미노 쌓기 프로젝트

2

프로젝트 내용

GUI를 사용하여 사용자에게 직접 경로를 제공받아 자동으로 각도를 인식, Manipulator를 제어하여 자동으로 도미노 쌓기 게임 진행

3

활용 장비 및 재료

Doosan Robot(M0609),
labtop,
ubuntu 22.04,
lan cable,
Tkinter

4

프로젝트 구조

도미노 쌓기 게임을 위해,
06월 09일 부터 06월
20일까지 프로젝트 수행
시스템 흐름 검토 및 통신
노드 구현, 프로그래밍 개발

5


활용방안 및 기대 효과

더 복잡한 task(피라미드
건축, 건물 쌓기 등)에
도입함으로써 자유로운 확장
가능

02

K-Digital Training

프로젝트 팀 구성 및 역할

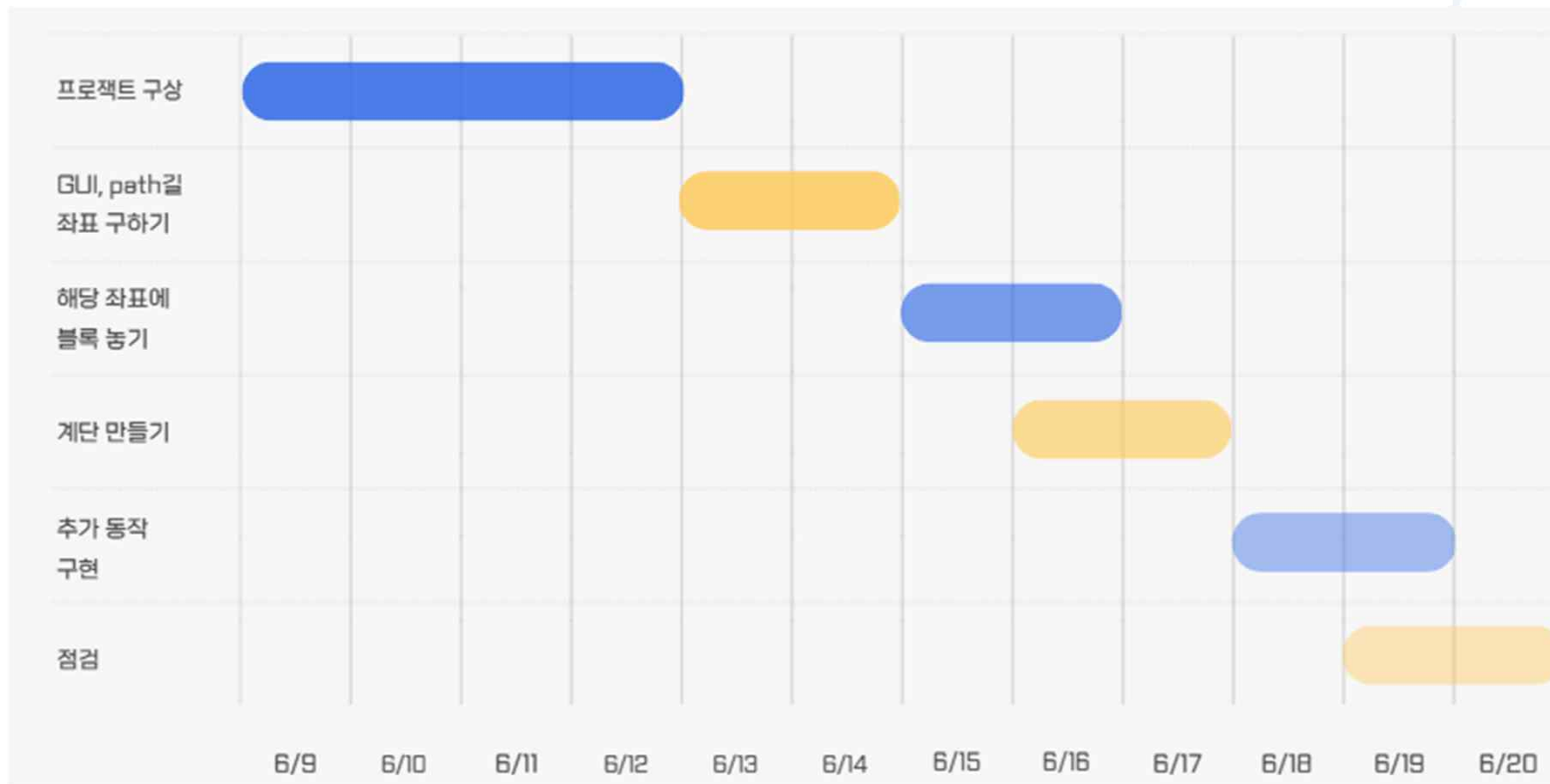
훈련생	역할	담당 업무	
한준모	팀장	 Robot Control	 Testing
배재성	팀원	 Extract Path	 GUI
김동호	팀원	 Testing	 Robot Control
김민수, 강현우	멘토	 주제 선정 피드백, 프로젝트 질의응답	

03

K-Digital Training

프로젝트 수행 절차 및 방법

▶ 프로젝트 수행 간트 차트



04

K-Digital Training

프로젝트 수행 경과

* 결과 제시 ① 탐색적 분석 및 전처리

▶ Introduction

- 최근 프로그래밍의 중요성 및 인기 확산으로 인한 많은 로봇(자율 주행, 로봇 팔, 4족 보행 등) 교보재 및 수업 등이 창설

홈 > 뉴스

홍대부중, 방과후 학교 통해 인공지능 코딩 교육 실시

▲ 박경일 | ⓒ 임력 2021.01.06 17:40 | ⓒ 수정 2021.01.06 19:10 | 댓글 0



로봇과 인공지능 결합 수업, 온라인으로 진행



▲ 홍익대학교 사범대학부속중학교에서는 방과후 학교와 동아리 활동으로 인공지능(AI) 교육이 한창 진행되고 있다.

Fig. 1 홍x중, 방과후 학교관련 기사

을 진행

로봇과학

신나고 재미있는 로봇과학



로봇과학은 단계별로 기초 조립부터 차근차근 있었고 파워 넘치는 다양한 로봇을 만들어 나가며 신나고 재미있는 배틀 시합과 미션, 게임, 경주를 통해 동작 원리를 배우고 논리적, 창의력, 집중력을 증대시킬 수 있습니다. 창작 작품을 제작하며 로봇에게 논리적, 순차적으로 올바르게 명령 프로그램을 작성하고 전달함으로써 프로그램의 기본원리를 이해하고 사고력, 문제 해결 능력이 풍부해질 수 있습니다.

선택형 (A~D반)	월	수	금	학년별 수업시간	교육비 30,000원 교재 및 교구비 99,000원~121,000원 (5-7개월)
A반	13:45~14:25	13:00~13:40	13:45~14:25	1~2학년	※ 로봇과학 월.수.금요일 수강인원 A반 1~2학년 20명 B반 1~4학년 20명 C반 1~6학년 20명 D반 1~6학년 20명
B반	14:35~15:15	13:50~14:30	14:35~15:15	1~4학년	
C반	15:20~16:00	14:40~15:20	15:20~16:00	전학년	
D반	16:05~16:45	15:30~16:10	16:05~16:45		

Fig. 2 전주송x초교 추가 수업

04

K-Digital Training

프로젝트 수행 경과

* 결과 제시 ① 탐색적 분석 및 전처리

▶ Introduction

- 하지만, 다양한 교보재 및 심지어 opensource 또한 복잡한 task는 예제에 존재하지 않음
- 창의성 향상을 위해서는 복잡한 task를 추가하여 생각의 범위를 넓힐 필요



Fig. 3 eduino판매 교육자료

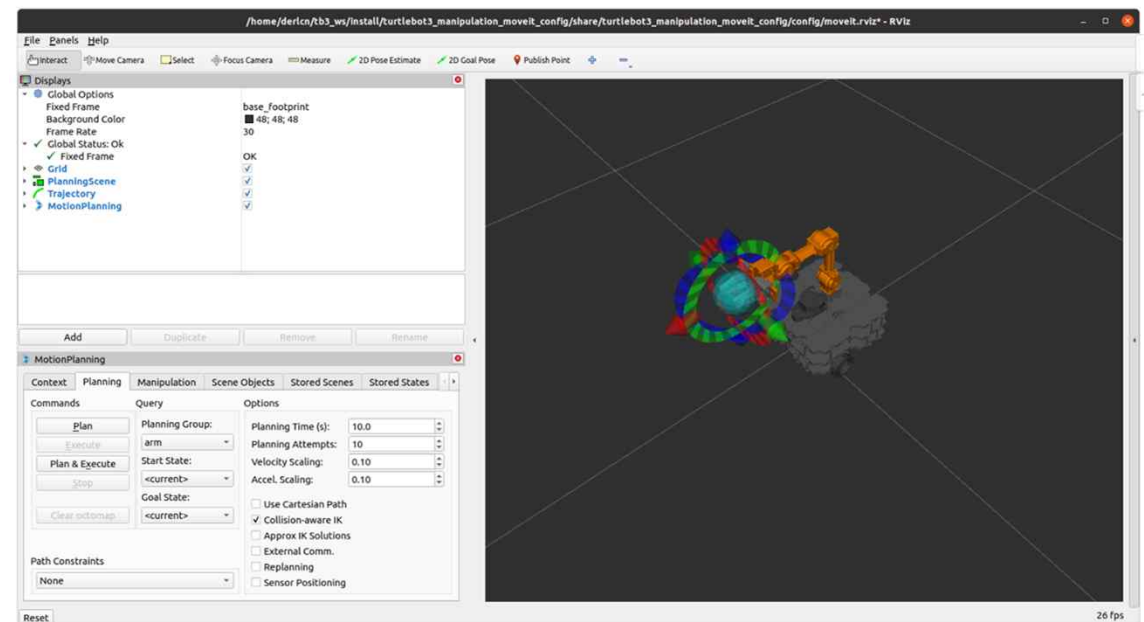


Fig. 4 turtlebot3 e-manual

04

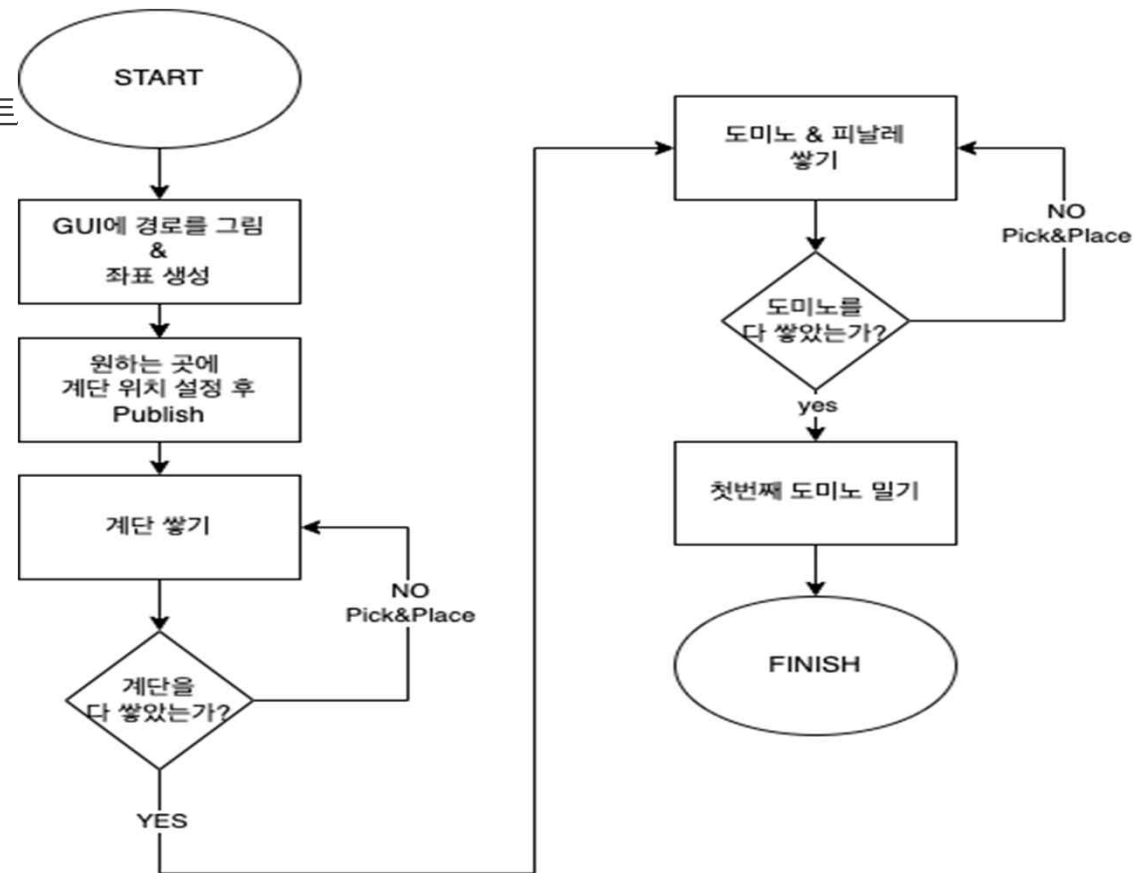
K-Digital Training

프로젝트 수행 경과

* 결과 제시 ② 모델 개요

▶ 전체 Flow Chart

- 사용자로부터 도미노 경로 입력, 도미노 쌓기 프로젝트
- GUI를 통한 사용자 경로 입력
- 샘플링후 yaw값 계산
- 계산된 정보를 Manipulator(M0609)로 publish
- 도미노 게임 제어



04

K-Digital Training

프로젝트 수행 경과

* 결과 제시 ② 모델 개요

▶ PoseArray 메시지에서 Yaw 값 추출

- draw_path에서 PoseArray를 구독
- Pose.orientation의 쿼터니언 값 추출
- Quaternion -> Euler변환
 - **roll, pitch, yaw**
- yaw값 활용 -> 도미노 회전
방향(rz)계산
- 계산된 rz값으로 도미 상태 제어

get_logger 중 일부

```
[cal_yaw-3] [INFO] [1750307111.405422749]
[path_to_posearray_node]:
geometry_msgs.msg.PoseArray(header=std_msgs.msg.Header(stamp
=builtin_interfaces.msg.Time(sec=1750307111, nanosec=391657095),
frame_id='base_link'),
poses=[geometry_msgs.msg.Pose(position=geometry_msgs.msg.P
oint(x=33.79999923706055, y=25.299999237060547, z=0.0),
orientation=geometry_msgs.msg.Quaternion(x=0.0, y=0.0, z=-
0.9383646691143729, w=0.34564685411250823)),
```

... (중략)

04

K-Digital Training

프로젝트 수행 경과

* 결과 제시 ② 모델 개요

▶ 도미노 계단 쌓기: 인덱스 & 층수 알고리즘

- draw_path에서 Int32MultiArray 구독
 - `[index, height]`
 - index = 몇번째 도미노?
 - height = 몇층?
- 계산의 높이 위치(z)를 계산
 - $z = \text{기준 높이} + (\text{층수} \times \text{단위높이})$
- 계산, 벽돌 쌓기, 피라미드 등 다양한 패턴 구현 가능

get_logger 중 일부

```
[cal_yaw-3] [INFO] [1750307111.405826608]  
[path_to_posearray_node]: Int32MultiArray (index,  
n_domino) published with 1 dominos  
[cal_yaw-3] [INFO] [1750307111.406092351]  
[path_to_posearray_node]:  
std_msgs.msg.Int32MultiArray(layout=std_msgs.msg.Multi  
ArrayLayout(dim=[], data_offset=0), data=[2, 3])
```

`[index, height]`

04

K-Digital Training

프로젝트 수행 경과

* 결과 제시 ② 모델 개요

▶ 도미노 마지막 피날레 자세 및 쓰러짐 구현

- 피날레 구현
 - 경로의 마지막 좌표를 50mm 연장
 - 마지막과 연장된 포즈 사이의 중간 좌표 계산
 - 수직으로 세워 피날레 구조를 생성
 - 계산의 높이 위치(z)를 계산
- 쓰러짐 구현
 - 도미시작 좌표 역방향 계산($\text{yaw} + \pi$)
 - manipulator를 역방향 30mm 만큼 오프셋 이동
 - 진행방향으로 물리적 push

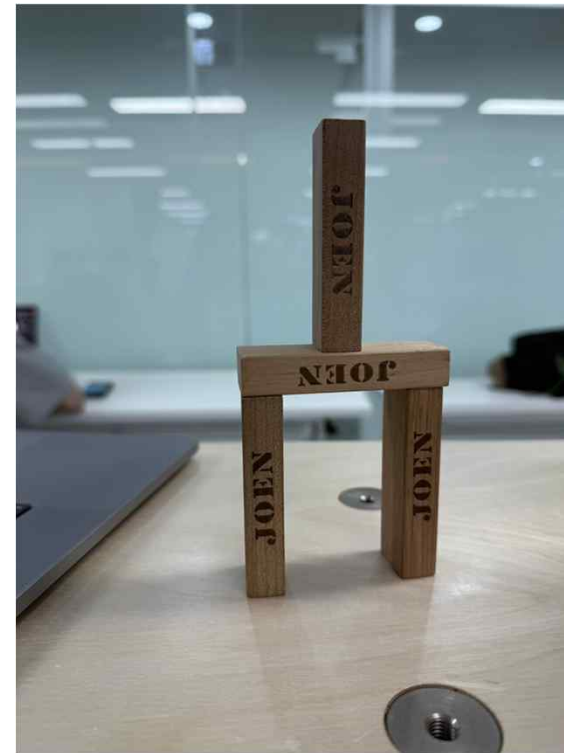


Fig. 6 Finale state

04

K-Digital Training

프로젝트 수행 경과

* 결과 제시 ③ 모델 선정 및 분석

▶ Path Extract Flow

- Tkinter를 활용해서 경로 그리기
- 등간격 sampling
- Yaw 계산 후, 로봇 제어로 publish



Fig. 7 Draw Path

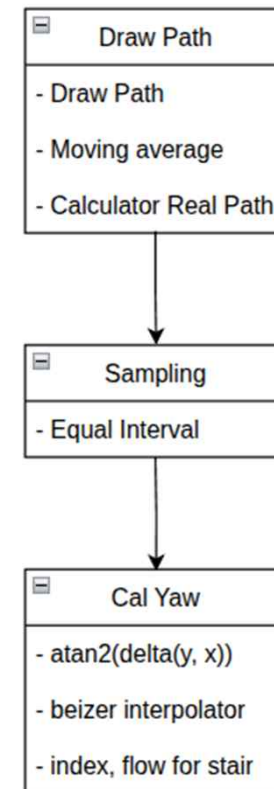


Fig. 8 Draw Path flow

04

K-Digital Training

프로젝트 수행 경과

* 결과 제시 ③ 모델 선정 및 분석

▶ Draw Path

- Function
 - Tkinter 활용 사용자에게 경로 입력
 - Moving Average
 - 사용자의 손떨림 방지
 - Pixel Coordi -> Real Coordi
 - Pixel Coordi(canvas) = 480 x 480 (pixel)
 - Real Coordi = 48 x 48 (cm)
 - 좌우 반전

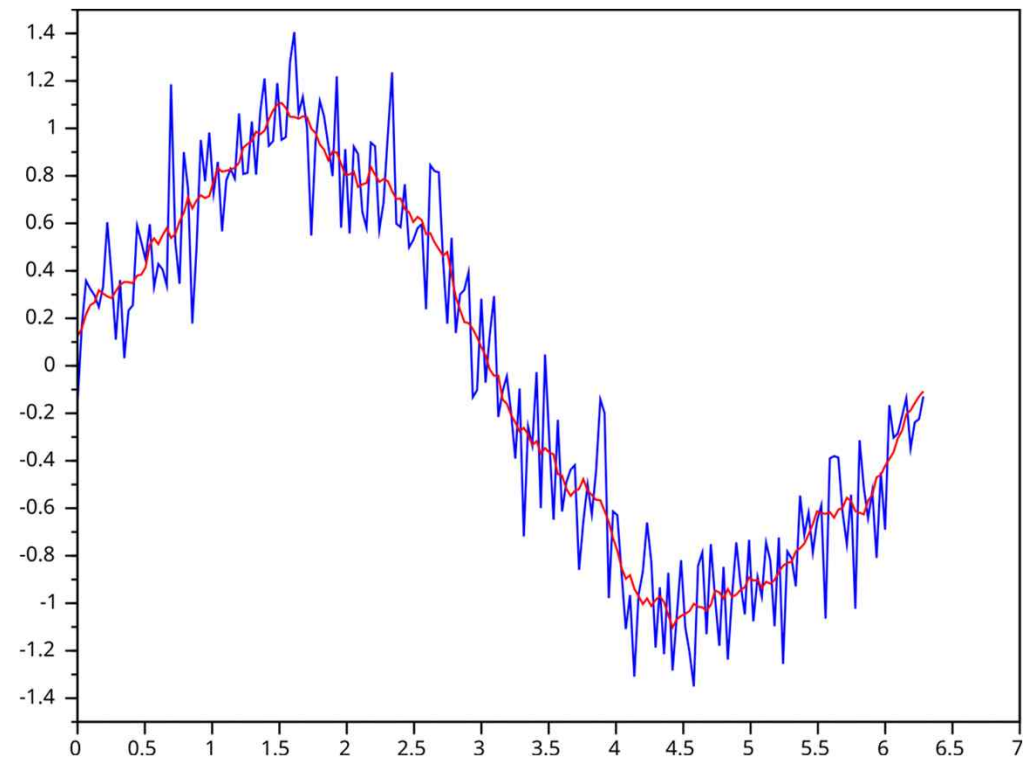


Fig. 9 Moving Average

04

K-Digital Training

프로젝트 수행 경과

* 결과 제시 ③ 모델 선정 및 분석

▶ Coordinate Sampling

- 등구간 샘플링

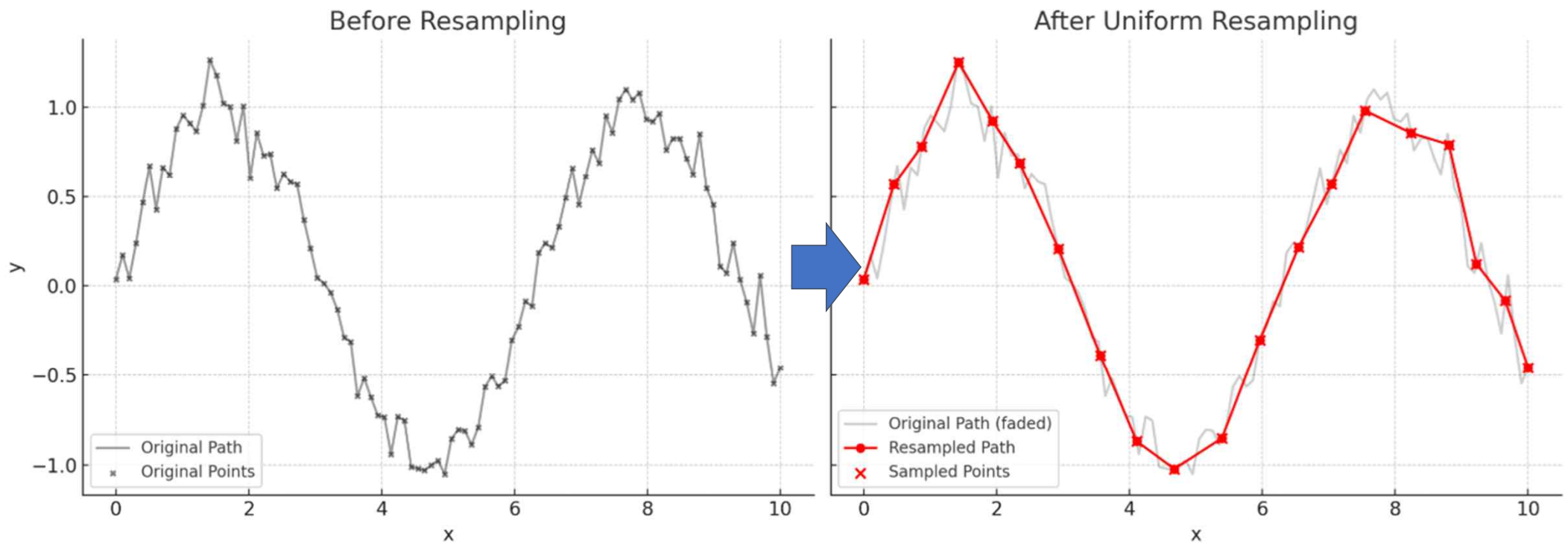


Fig. 10 Sampling

04

K-Digital Training

프로젝트 수행 경과

* 결과 제시 ③ 모델 선정 및 분석

▶ Calculator Yaw

- Function
 - delta y, x를 활용한 yaw 계산(라디안)
 - $z = 150$ 일때, roll, pitch, yaw 고려
 - Tkinter 활용 샘플링 및 yaw 계산결과 가시화
 - 좌우반전
 - 계산 쌓기를 위한, 사용자 입력 대기 및 publish
 - 계산 작업을 진행할 좌표 인덱스
 - 계단 층수
 - PoseArray publish

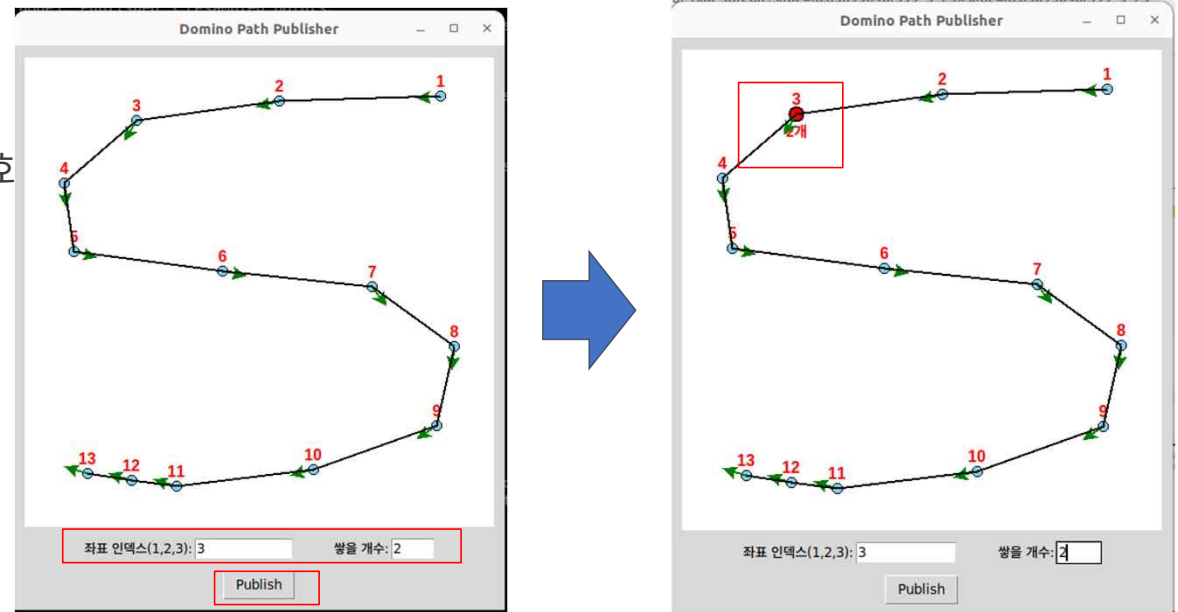


Fig. 11 Stair index & floor

04

K-Digital Training

프로젝트 수행 경과

* 결과 제시 ③ 모델 선정 및 분석

▶ Calculator Yaw

- Function
 - delta y, x를 활용한 yaw 계산(라디안)
 - z = 150 일때, roll, pitch, yaw고려
 - **Interpolation(보간)**
 - **Catmull-Rom Splines vs Bezier**
 - Tkinter 활용 샘플링 및 yaw 계산결과 가시화
 - 좌우반전
 - 계산 쌓기를 위한, 사용자 입력 대기 및 publish
 - 계산 작업을 진행할 좌표 인덱스
 - 계단 층수
 - PoseArray publish

04

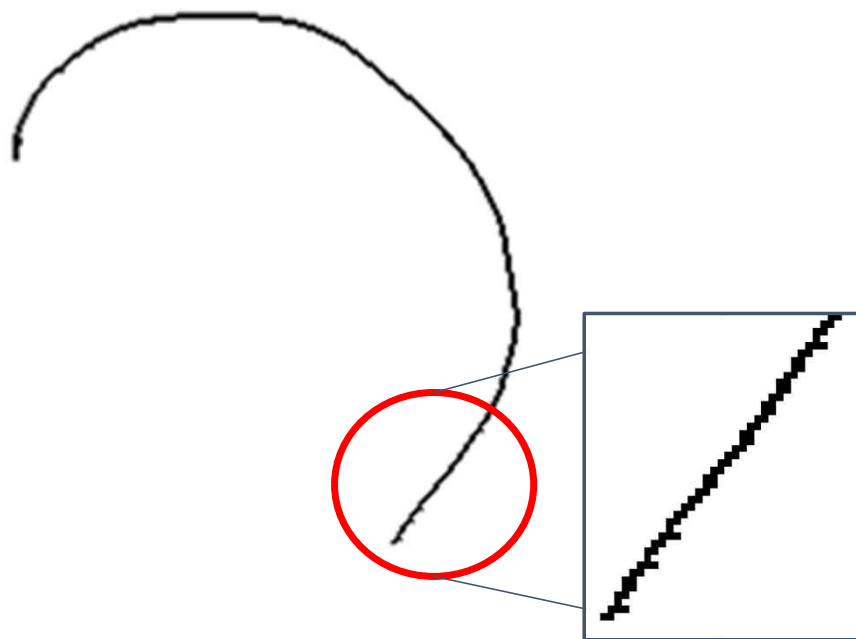
K-Digital Training

프로젝트 수행 경과

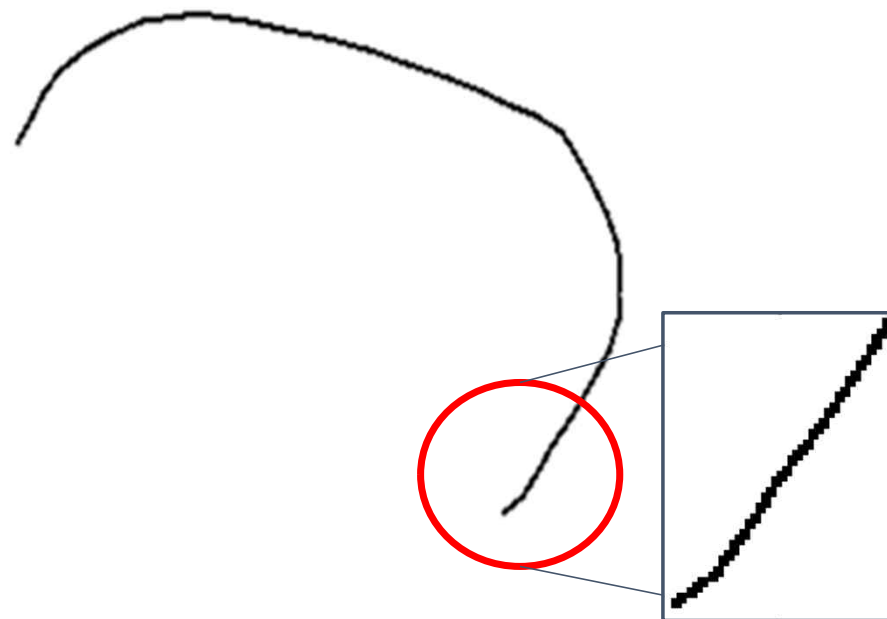
* 결과 제시 ③ 모델 선정 및 분석

▶ Calculator Yaw

- Catmull-Rom Splines 결과



- Beizer Splines 결과



04

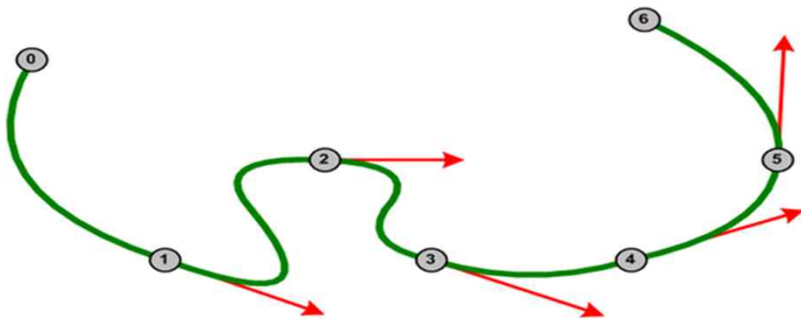
K-Digital Training

프로젝트 수행 경과

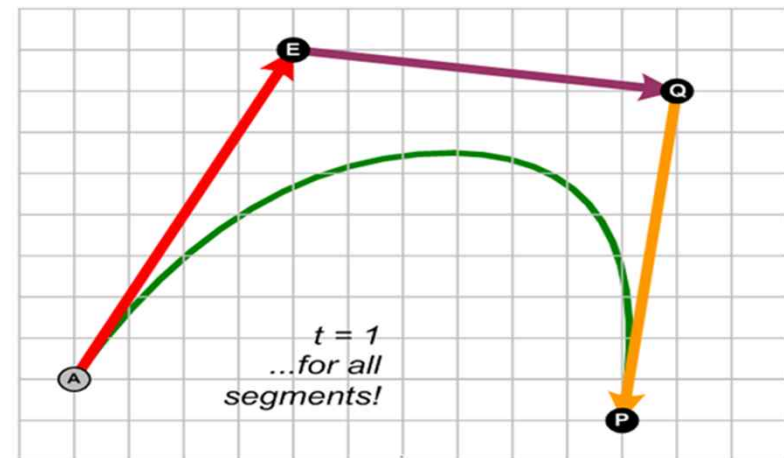
* 결과 제시 ③ 모델 선정 및 분석

▶ Calculator Yaw

- Catmull-Rom Splines 원리



- Beizer Splines 원리

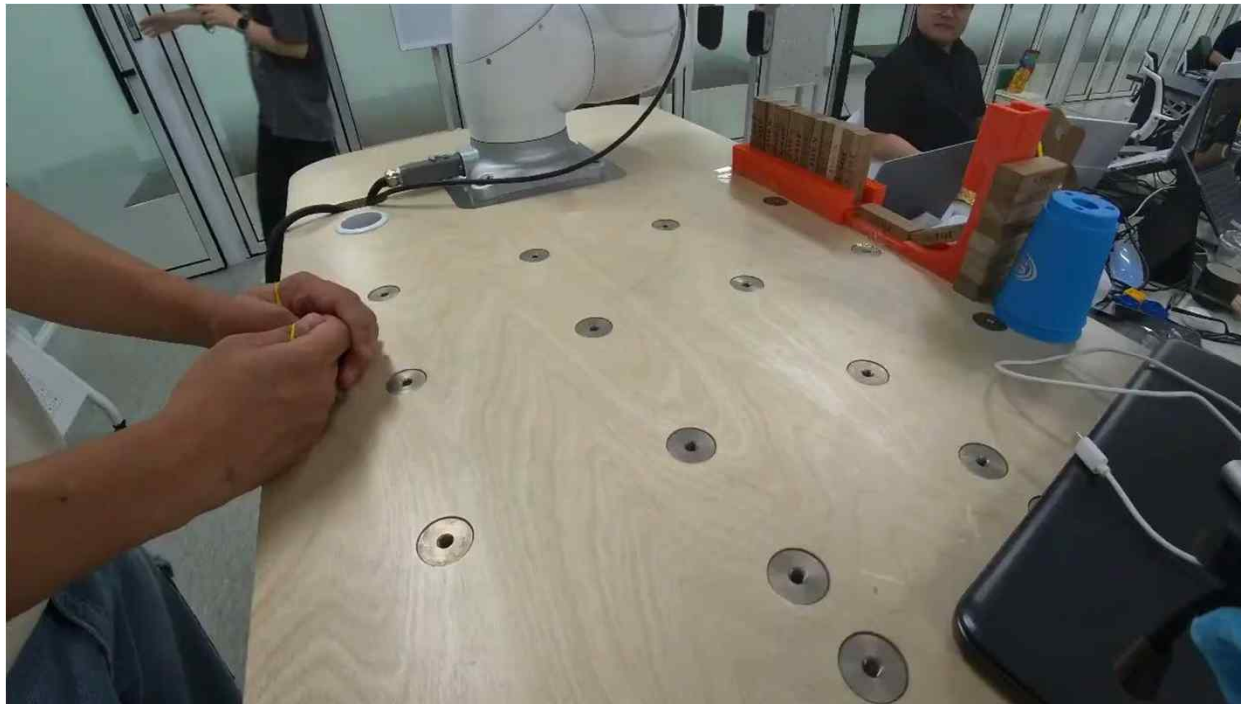


04

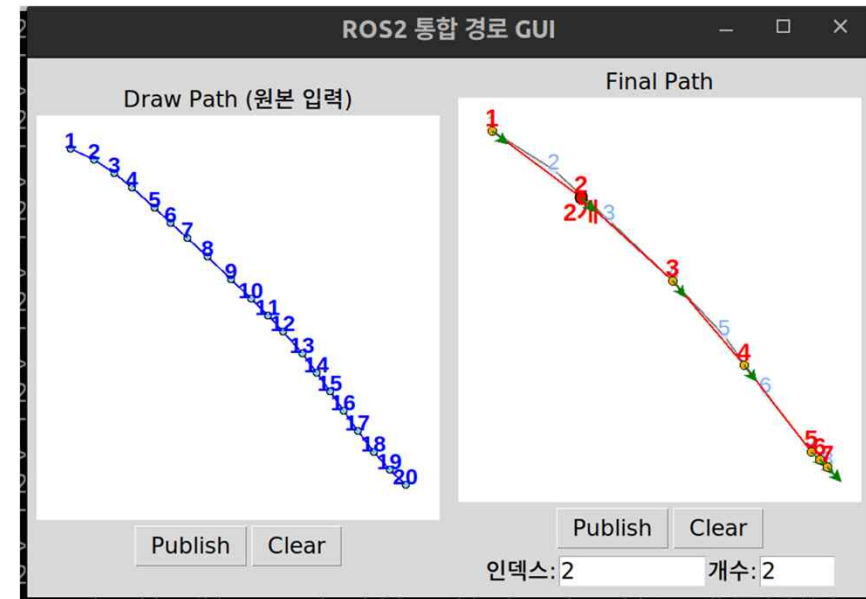
K-Digital Training

프로젝트 수행 경과

▶ 원테이크 영상



* 결과 제시 ③ 모델 선정 및 분석



05

K-Digital Training

자체 평가 의견

프로젝트 결과물에 대한 완성도 평가

(10점)

- 프로젝트에 필요한 기능 및 수식을 적절히 활용
- 높은 두산 API 이해도

추후 개선점이나 보완할 점

- 복잡한 경로(곡선) 가능
- yaw값 계산 방식 : 수식이 아닌 URDF기반 계산

잘한 부분과 아쉬운 점

- 다양한 보간법 적용
- 절적인 힘제어 적용(높이 고려 가능)
- yaw값에 대한 결과 오차 발생

느낀 점이나 경험한 성과

- 로봇팔에 대한 기능들을 동작시켜보며 경험할 수 있는 좋은 기회였음