

디지털 트윈 기반 서비스 로봇 운영 시스템 구성

디지털 트윈 기반 사과 선별 공정 자동화

협동 -3

AI & Digital Twin 기반 지능형 자동화 시스템

Project Team

F-2 삼성

이승민, 이주학, 최대혁

멘토 : 김민수, 김도영

목 차 INDEX

사과 선별 공정 디지털 트윈

F-2 삼성

01

프로젝트 개요

주제 · 배경 · 비전 시스템 · 구조 · 활용방안



02

팀 구성 및 역할

팀원별 담당 업무 및 협업 구조



03

핵심 전략

파라미터 스윕 최적화 및 기술적 접근 전략



04

수행 경과

데이터 확장 · Split 개선 · 증강 전략 (모델 최적화 스토리)



04-1

핵심 기술 및 통신 검증

코드 흐름 · DB 저장 · ROS2 통신 파이프라인 검증



05

사후 평가

완성도 평가 · 개선점 · 소감



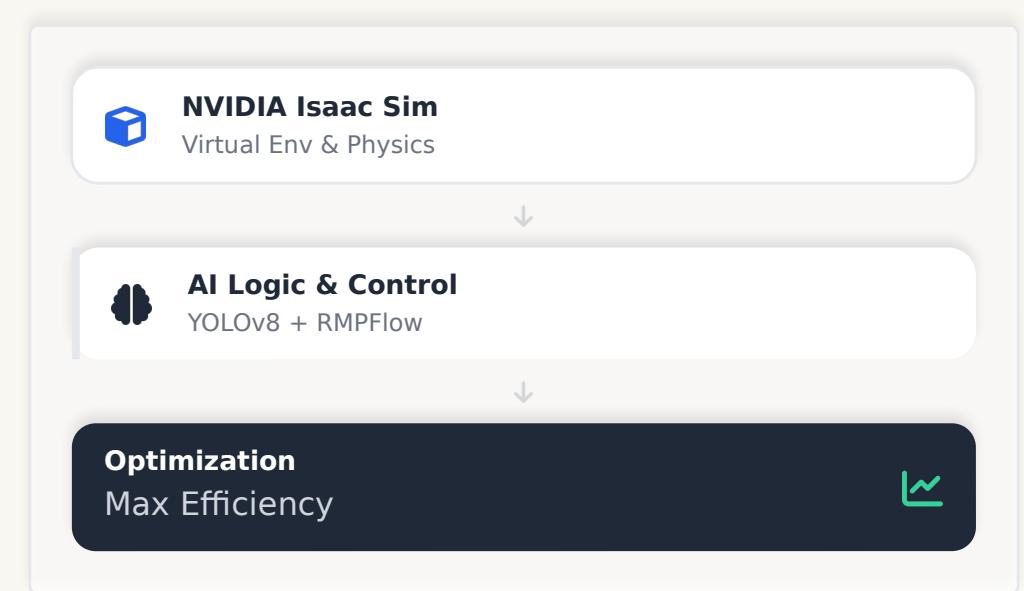
프로젝트 개요 주제 및 기획 의도

주제 : AI & 디지털 트윈 융합 자동화

AI 비전 (YOLO) 과 Isaac Sim 물리 제어 기술을 결합하여
사과 선별 공정의 완전 자동화 및 가상화 시스템을 구축합니다.

기획 의도 : 가상 검증을 통한 리스크 최소화

설비 구축 전 가상 환경 (Sim) 에서의 사전 검증을 통해
현장 도입 시 발생하는 비용 낭비와 다운타임 (Downtime) 을 획기적으로 제거합니다.



디지털 트윈 설계

가상 실험 (Sim)

성능 /ROI 예측

현장 적용

프로젝트 개요 선정 배경 및 문제 정의

육안 선별의 한계와 품질 편차

사과의 계절 · 품종 · 외형 편차와 작업자 피로도로 품질 불균형 발생

설비 변경 시 비용 리스크

파라미터 변경 시 **라인 중단 (Downtime)** 과 막대한 실험 비용 불가피

현장 반복 실험의 물리적 제약

비정형 상황 재현 불가 → 시뮬레이션 기반 **가상 파라미터 탐색** 필수

Reality (Pain Points)



중단



비용



불가

Digital Twin Solution

⚡ 무한 반복 테스트

☞ 파라미터 즉시 최적화

↪ 사전 검증 ROI 예측

Necessity Flow

현장 제약 증가

반복 실험 난해

비용 / 리스크

디지털 트윈 도입

프로젝트 개요 내용 (공정 범위)

주제 : 협동 -3 디지털 트윈 기반 서비스 로봇 운영 시스템

팀명 : F-2 삼성

인피드 및 라인 정렬

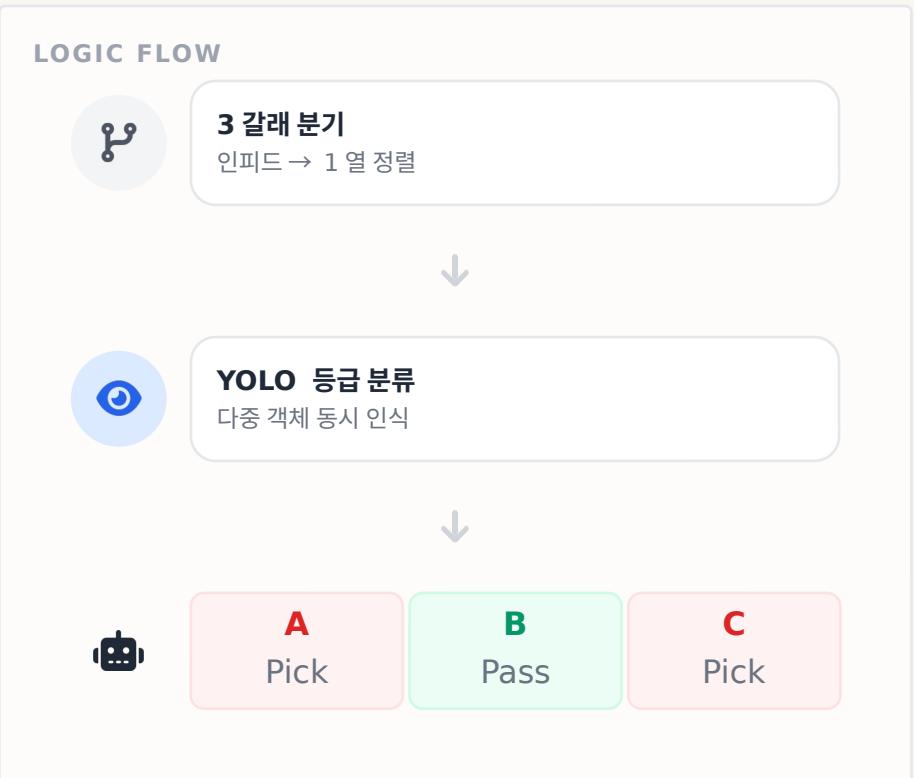
인피드 광폭 투입 → 3 갈래 분기 → 1 열 정렬 유도

비전 인식 및 로봇 제어

YOLO 다중 검출로 등급 분류후,
로봇이 A·C 등급은 픽 (Pick) 하고 B 등급만 통과 (Pass) 하여 배출

시뮬레이션 제작 및 최적화

컨베이어는 형상만 렌더링하고, 실제 이동은
사과 속도 · 굴림 모델로 경량화하여 성능 확보



Process Flow Architecture



프로젝트 개요 구조 및 활용 방안

주제 : 협동 -3 디지털 트윈 기반 서비스 로봇 운영 시스템

팀명 : F-2 삼성

모듈러 시스템 구조 (Modular Architecture)

전체 공정을 **인피드 / 분기 / 비전 / 분류 / 로봇 / 배출**의
독립된 모듈로 설계하여 유지보수와 기능 확장이 용이한 구조 확립

활용 방안 : 사전 검증 및 최적화

디지털 트윈 환경에서 충분히 테스트.
“빠르게 처리할 것인가, 정확하게 분류할 것인가”에 대해 현장에 맞는 최적의 균형점 도출

확장성 (Scalability)

사과 선별 외에도 유사한 외형을 가진 **부품 (볼트 / 너트)**이나
다른 농산물 (감귤, 토마토 등)의 등급 분류 공정으로 범용화 가능

Development Workflow

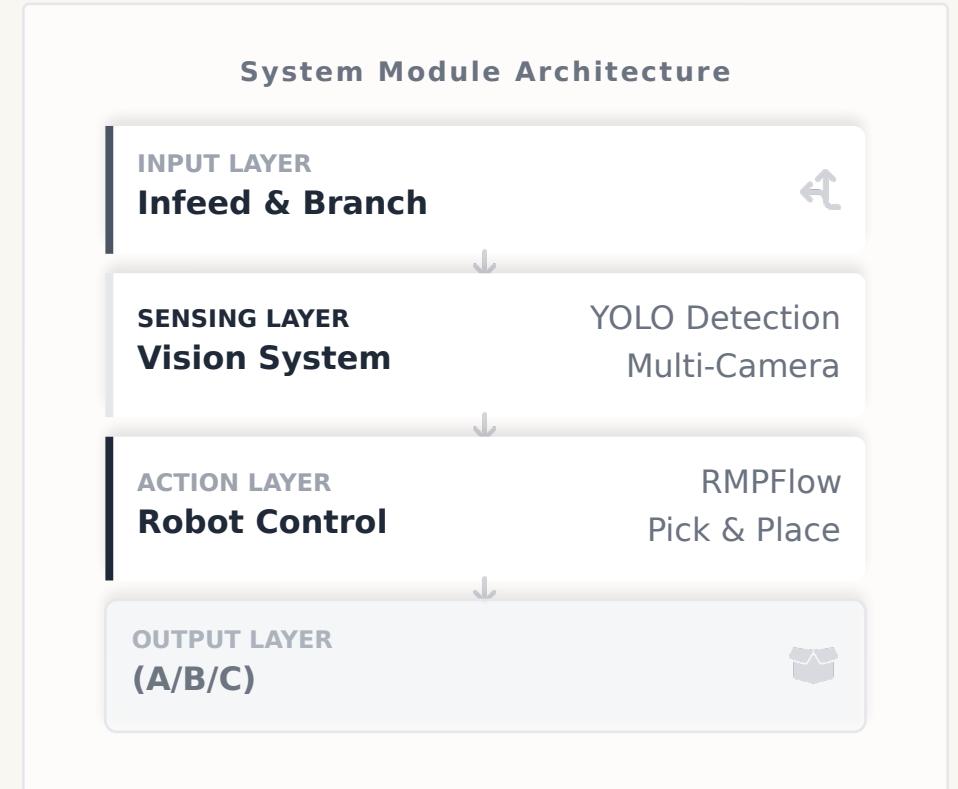
시스템 설계

시뮬레이션 구현

파라미터 튜닝

결과 리포트

현장 전개



프로젝트 전체 구조 및 공정 흐름 도식

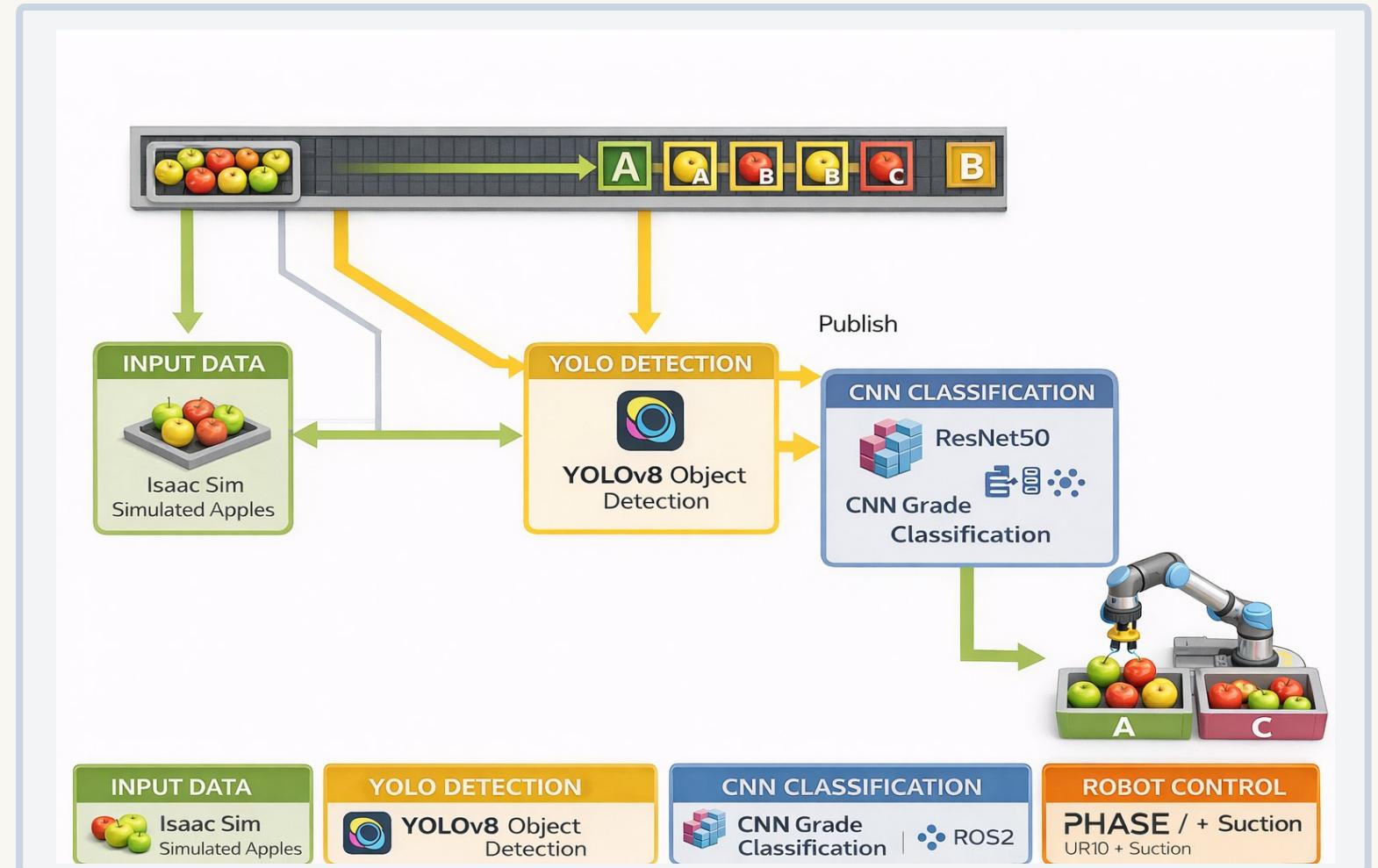
팀명 : F-2 삼성

모듈러 시스템 구조

- 각 공정 단계별 독립적 기능 수행 고려
- 유연한 확장성을 고려한 아키텍처 설계

전체 공정 파이프라인

인피드 분기 비전 분류 로봇 배출

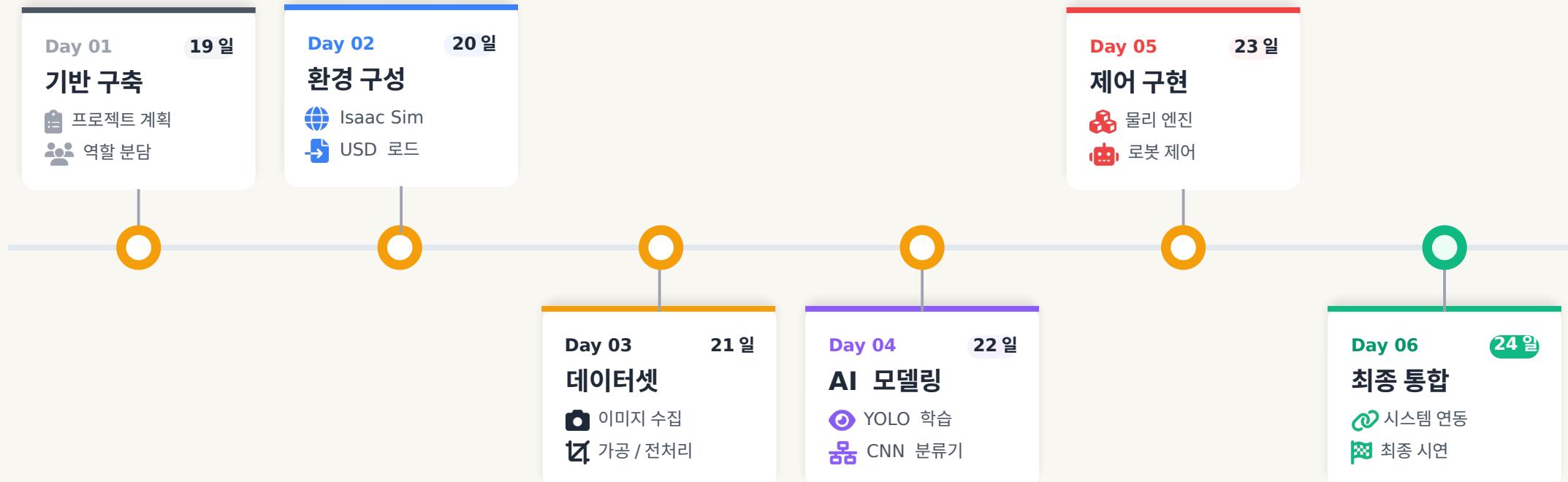


프로젝트 수행 절차 및 마일스톤

F-2 삼성

Start: 19 일

End: 24 일



데이터 구축

Digital Twin

AI & Logic

검증 완료

사과 투입 및 물리 엔진 적용 결과

주제 : 협동 -3 디지털 트윈 기반 서비스 로봇 운영 시스템

팀명 : F-2 삼성

| 핵심 검증 포인트

물리 엔진 최적화 주요 성과



물리엔진 충돌처리

Collision Handling



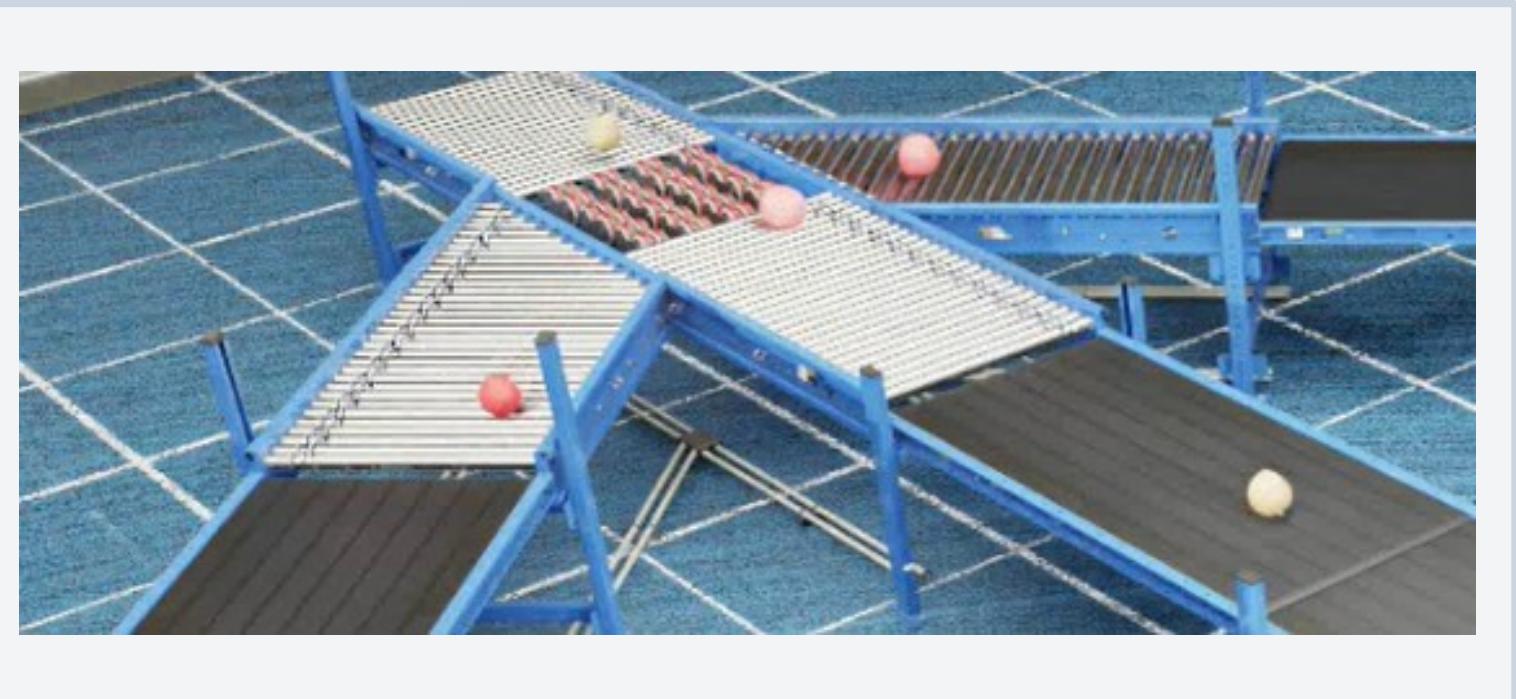
병목 현상 방지

Bottleneck Prevention



안정적 동작 구현

Stable Operation



비전 시스템 최적화 카메라 선정 검증

▣ 카메라 각도 비교 실험

사과 선별 정확도를 높이기 위해 두 가지 카메라 배치 (상부 vs 사선)에 따른
객체 인식률 (mAP) 차이를 비교 분석하였습니다.

💡 실험 결과 및 분석

구 (Sphere) 형태 특성 : 각도 변화보다 조명 및 해상도가 더 큰 변수

모델 정확도 차이 없음

사과는 둥근 형태로, 어느 각도에서 촬영하든 외형적 특징 (Shape Features) 차이가 미미함

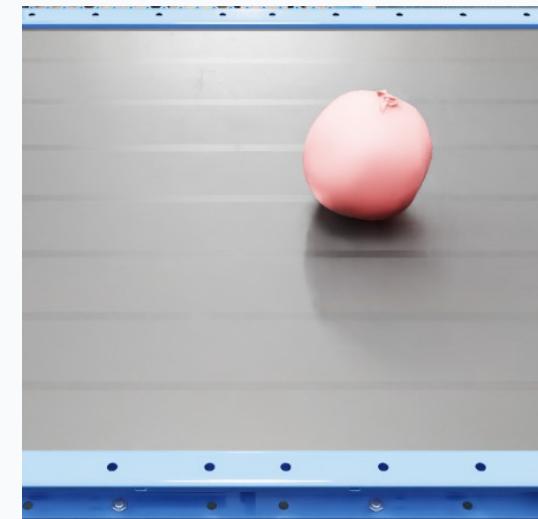
✓ 최종 선정 : 상부 단일 카메라

복잡한 멀티 카메라 구성 대신 **상부 단일 카메라**를 선택하여 시스템 복잡도를 낮추고
데이터 처리에 집중했습니다.

▣ 카메라 각도 실험

└ mAP 비교 분석

Case A



Case B



“

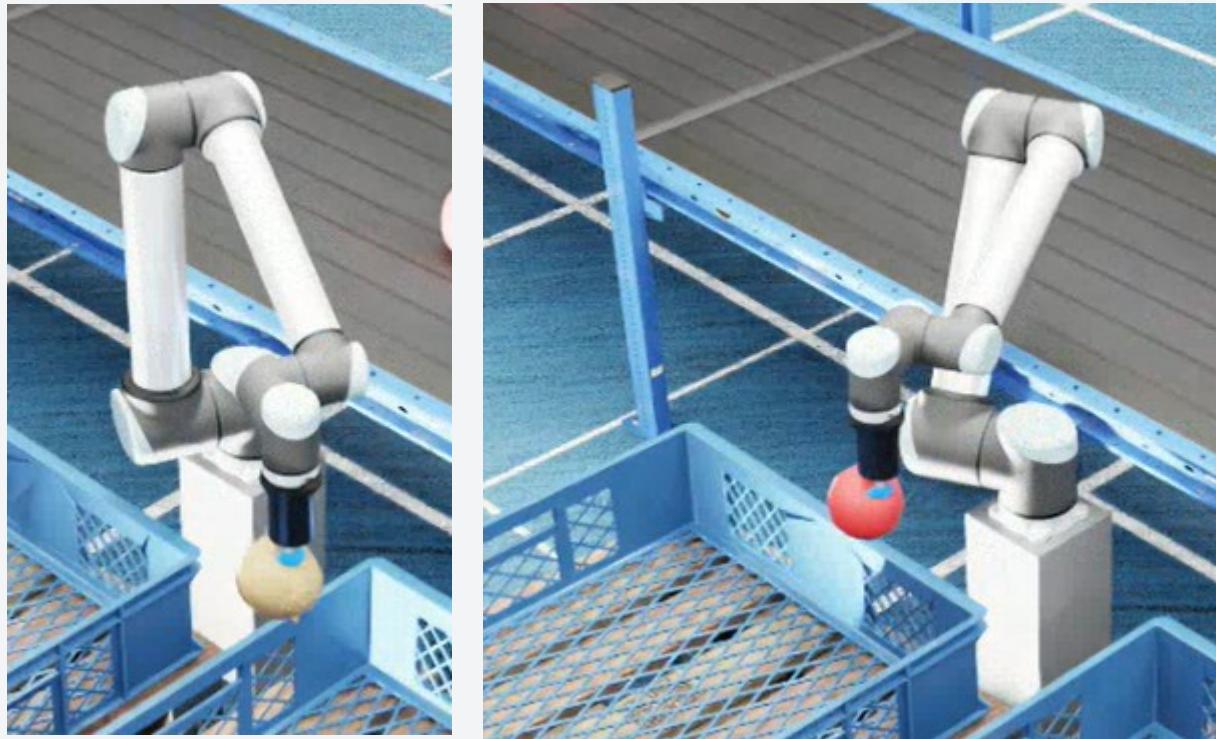
“각도보다 데이터 · 증강 · 학습 전략이
성능을 결정합니다.”

물리적 카메라 배치보다 데이터셋 품질 확보 (Data-Centric AI) 가 핵심임을 입증

✓ 상부 단일 채택

└ 데이터 중심 최적화

로봇 암 Pick & Place



핵심 동작 포인트

01 PHASE 기반 제어

안정적인 상태 관리 로직 적용

02 Suction Open-Close

정확한 진공 흡착 / 해제 타이밍

03 Pick & Place 성공

오차 없는 목표 좌표 이동

RGB 카메라와 로봇 암 좌표계 동기화를 통해 정확한 위치 제어를 수행합니다.

팀 구성 및 역할 Team Structure

Project F-2 삼성

이승민

TEAM
LEADER



시스템 설계 및 모델 구축

- **YOLO-CNN** 모델 구축
- 전체 시스템 설계 및 통합
- 학습 데이터 정제

이주학

MEMBER

환경 구성 및 디자인

- 디지털 트윈 환경 구성
- 시나리오 디자인
- 공정 시뮬레이션 설계

최대혁

MEMBER

데이터 수집 및 기획

- 디지털 트윈 환경 구성
- 발표 자료 (PPT) 제작
- 물리 디자인

수행 절차 및 방법

주제 : 협동 -3 디지털 트윈 기반 서비스 로봇 운영 시스템

팀명 : F-2 삼성

◎ 핵심 전략

• 파라미터 스윕 (Sweep) 최적화

카메라 위치, 컨베이어 속도, 분기 각도 등 주요 변수를 가상 환경에서 반복 실험하여 최적값 도출

• 시뮬레이션 기반 병목 제거

3D CAD 분기 지점과 로봇 작업 영역의 트래픽을 분석하여 병목 구간 사전 식별 및 해소

Implementation Workflow



기획 및 시나리오

공정 설계, 요구사항 정의, 시나리오 구체화



데이터 준비

사과 / 과일 USD 모델링, 컨베이어 에셋 확보



환경 구축

Isaac Sim 씬 구성, 조명 및 카메라 배치



통합 검증

전체 공정 연동 테스트, 성능 지표 확인



로봇 연동

RMPFlow 모션 제어, Pick & Place 로직



AI / 물리 결합

YOLO 분류 모델 적용, 물리 엔진 (충돌) 튜닝

PROJECT TOOLS

Isaac Sim Python YOLOv8 ROS2

Overall Process Flow

기획

데이터

환경

물리 + AI

로봇

통합 검증

1 Problem

문제 인식 (초기 상태)

초기 데이터 수

22 장 (매우 부족)

✗ 클래스 간 미세한 차이 (A/B/C) 분류 불가

3 Distribution

분할 방식 개선

기존 고정 폴더 분리 → 편향 (Bias)

개선 Random Split 적용 → 일반화 성능 ↑

Train (Random)

Val (Random)

5 Training

학습 전략 변화

Epoch Flow

20 → 7 → 2

✓ 데이터 품질 확보 시 작은 Epoch 로 고성능 달성

2

Quantity

데이터 확장 전략

확장 규모

20 장 → 200 장 +

💡 단순 증량이 아닌 '형태 다양성 (회전 / 위치 / 가림)' 확보가 핵심 목적

4

Augmentation

증강 전략 조정

Maintained

Flip Rotate

Remove
Random Crop

Why? 사과 등급의 핵심인 전체 형태 보존 필수

6

Conclusion

최종 효과 요약

- ✓ 데이터 수 확장 + 증강 최적화
- ✓ Random Split 검증 안정성 확보
- ✓ 작은 Epoch 으로 고성능 달성

모델 최적화 증강 전략 및 Hyperparameter 튜닝

증강 전략 변경 : Random Crop 제거

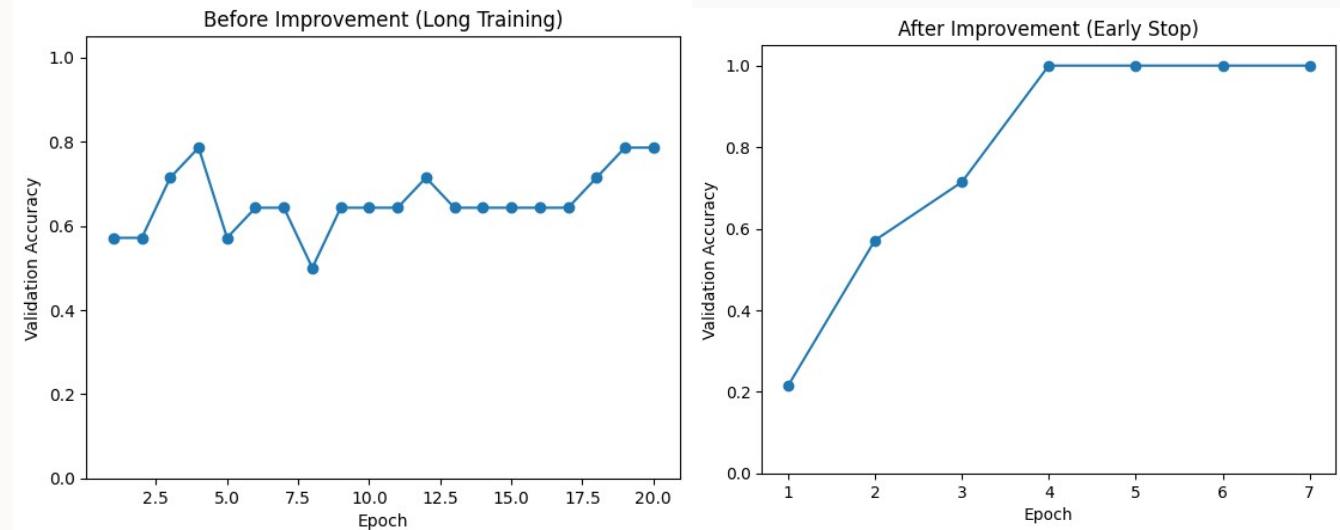
사과의 형태 (꼭지, 돌기) 가 등급 분류의 핵심 단서입니다.
Random Crop은 이를 잘라내어 성능 저하를 유발하므로 제거하고,
형태 보존 위주의 증강 (Rotation, Flip 등)으로 전환했습니다.

Epoch 튜닝 ($20 \rightarrow 7 \rightarrow 2$) 및 결론

초기 Epoch 20에서 과적합(Overfitting) 징후 발생

데이터 품질 확보 후 Epoch 2만으로도 최고 성능 달성

결론 : 충분한 양의 양질의 데이터 확보 시 과도한 Epoch 학습 불필요



```
rokey@rokey-Vector-16-HX-AI-A2XWIG:~/Downloads/work_23$ python3 train_resnet50_abc.py
RUNNING FILE: /home/rokey/Downloads/work_23/train_resnet50_abc.py
DATA_ROOT: /home/rokey/Downloads/work_23
device: cuda
classes (MUST be ['A', 'B', 'C']): ['A', 'B', 'C']
train_len: 649 val_len: 162
class_counts(train): [206, 221, 222]
class_weights: [0.004854368977248669, 0.004524887073785067, 0.004504504613578319]
unique targets: [0, 1, 2]
Epoch 1/2: 100%[██████████] 21/21 [00:05<00:00,  3.64it/s, loss=0.0142, lr=0.0003]
[VAL] loss=1.0702 acc=0.7469
[SAVE:BEST] resnet50_abc_out/best_resnet50_abc.pt (acc=0.7469)
[SAVE:CKPT] resnet50_abc_out/checkpoint_latest.pt (epoch=1)
Epoch 2/2: 100%[██████████] 21/21 [00:05<00:00,  3.71it/s, loss=0.0314, lr=0.00015]
[VAL] loss=0.0000 acc=1.0000
[SAVE:BEST] resnet50_abc_out/best_resnet50_abc.pt (acc=1.0000)
[SAVE:CKPT] resnet50_abc_out/checkpoint_latest.pt (epoch=2)
[SAVE:LAST] resnet50_abc_out/last_resnet50_abc.pt
DONE best_acc: 1.0
```

증강 전략 수정

Random Crop 제거

Epoch 단축 ($20 \rightarrow 2$)

최적 모델 확보

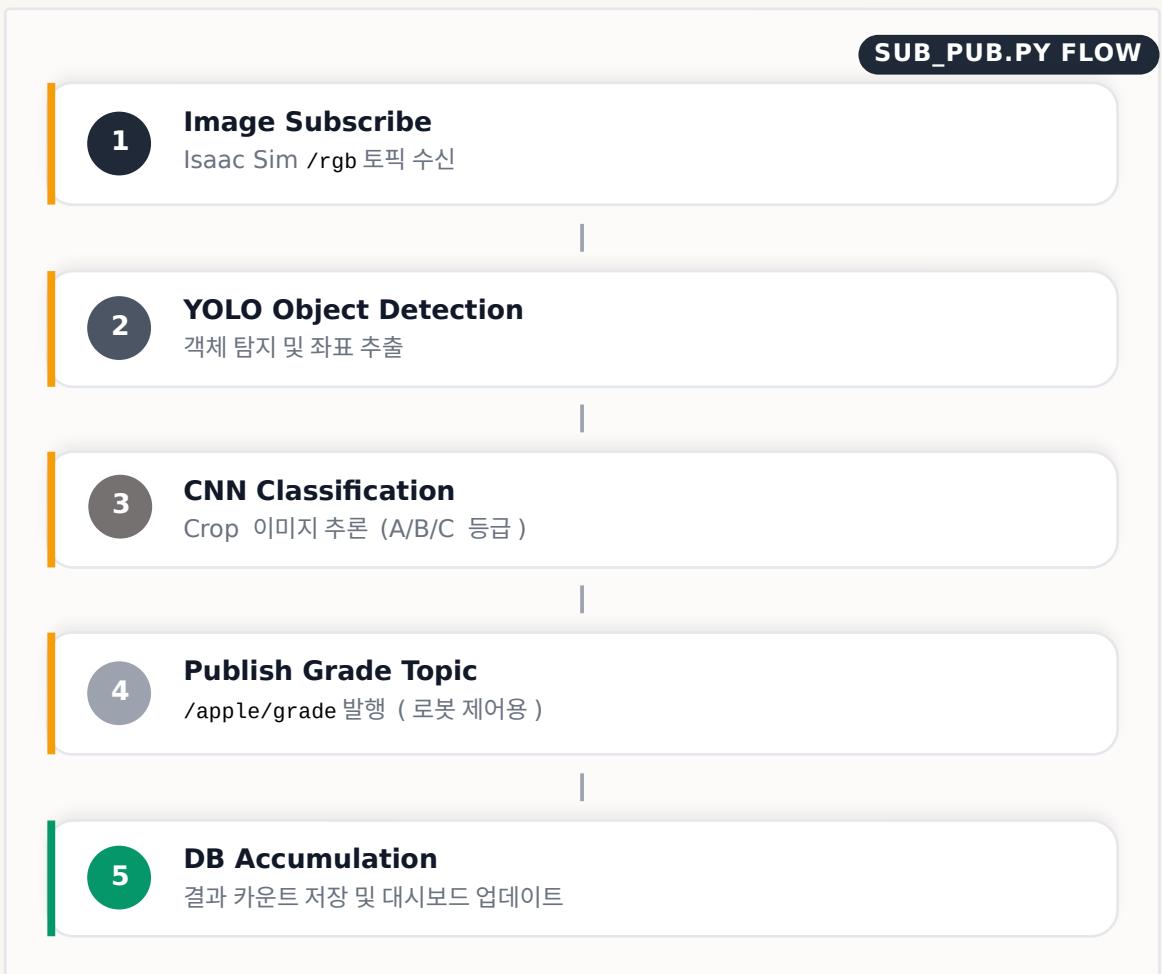
DB 기반 결과 관리 및 통신 검증

> ROS2 Topic Publish Log

```
[INFO] [1769240551.835480867] [camera_listener]: Subscribing: /rgb
[INFO] [1769240551.835995789] [camera_listener]: Publishing: /apple/grade
[INFO] [1769240551.836103121] [camera_listener]: DB: /home/rokey/Desktop/detect_classification/apple_grade_counts.db
[INFO] [1769240600.496003560] [camera_listener]: PUBLISH /apple/grade: C (conf=1.000) | box
=0 | pub_count=1 | DB C=1 | ALL={'A': 0, 'B': 0, 'C': 1}
[INFO] [1769240616.214098367] [camera_listener]: PUBLISH /apple/grade: B (conf=1.000) | box
=0 | pub_count=2 | DB B=1 | ALL={'A': 0, 'B': 1, 'C': 1}
[INFO] [1769240629.691559004] [camera_listener]: PUBLISH /apple/grade: B (conf=1.000) | box
=0 | pub_count=3 | DB B=2 | ALL={'A': 0, 'B': 2, 'C': 1}
[INFO] [1769240646.308008897] [camera_listener]: PUBLISH /apple/grade: A (conf=1.000) | box
=0 | pub_count=4 | DB A=1 | ALL={'A': 1, 'B': 2, 'C': 1}
[INFO] [1769240660.631695992] [camera_listener]: PUBLISH /apple/grade: B (conf=1.000) | box
=0 | pub_count=5 | DB B=3 | ALL={'A': 1, 'B': 3, 'C': 1}
[INFO] [1769240675.987659398] [camera_listener]: PUBLISH /apple/grade: B (conf=1.000) | box
=0 | pub_count=6 | DB B=4 | ALL={'A': 1, 'B': 4, 'C': 1}
[INFO] [1769240691.571523787] [camera_listener]: PUBLISH /apple/grade: B (conf=1.000) | box
=0 | pub_count=7 | DB B=5 | ALL={'A': 1, 'B': 5, 'C': 1}
[INFO] [1769240706.387894051] [camera_listener]: PUBLISH /apple/grade: C (conf=0.864) | box
=0 | pub_count=8 | DB C=2 | ALL={'A': 1, 'B': 5, 'C': 2}
[INFO] [1769240722.839505002] [camera_listener]: PUBLISH /apple/grade: B (conf=1.000) | box
=0 | pub_count=9 | DB B=6 | ALL={'A': 1, 'B': 6, 'C': 2}
[INFO] [1769240736.695170106] [camera_listener]: PUBLISH /apple/grade: A (conf=1.000) | box
=0 | pub_count=10 | DB A=2 | ALL={'A': 2, 'B': 6, 'C': 2}
```

Classification Results DB

SUB_PUB.PY FLOW



검증 포인트 : 분류 (Inference) → 통신 (Publish) → DB 저장 (Archive) 까지 데이터 파이프라인 무결성 확인

핵심 기술 및 코드 흐름

sub_pub.py / train_resnet50_abc.py

Core Logic Details



데이터셋 구축 및 학습

YOLOv8로 객체 (사과) 를 자동 감지하여 이미지를 저장하고 , **train_resnet50_abc.py** 에서 CNN 모델을 학습시켜 Best 가중치 생성



ROS2 통신 브리지 (sub_pub.py)

Isaac Sim에서 전송하는 **/rgb** 토픽을 구독 (Sub) 하고 , 분류 결과를 **/apple/grade** 토픽으로 발행 (Pub) 하여 시뮬레이션 제어



Hybrid Inference (2-Stage)

1 단계 : **YOLO** 가 이미지 내 사과 위치 (BBox) 탐지
2 단계 : **ResNet50** 이 A/B/C 등급 정밀 분류



분류 기반 로봇 제어

판정된 등급 (A/C) 은 **Pick & Place** 수행 , B 등급은 통과 (Pass) 시키는 로직을 물리 엔진에 실시간 반영

EXECUTION FLOW

1

Data Collection

YOLO 객체 감지 → Image Save

2

Model Training

ResNet50 CNN 모델 학습 → Best 가중치 저장

3

Real-time Inference

Isaac Sim RGB 토픽 수신 → YOLO+CNN 등급 분류

4

Publish Result

ROS2 Topic 발행 (/apple/grade: A/B/C)

5

Robot Control

분류 결과 기반 Pick(A/C) or Pass(B) 동작

사후 평가 완성도 자체 평가

핵심 기술 구현 성공 (Core Technology)

AI 기반 자동 등급 분류 성공 : YOLO + CNN 모델 융합으로 정밀도 확보

디지털 트윈 환경 구축 완료 : Isaac Sim 기반 가상 공정 구현

시스템 통합 및 데이터 관리 (Integration)

ROS2 연동 로봇 암 제어 안정화 : Pick & Place 동작 최적화

DB 기반 분류 결과 관리 시스템 구현 : 실시간 데이터 파이프라인 완성

Overall Score

Target: 10

10 / 10

Achievement

100%

사전 기획 단계에서 설정한
핵심 기능 구현 및 시스템 통합 목표를
성공적으로 달성하였습니다.

설계 (Design)

구현 (Dev)

검증 (Test)

평가 (Evaluation)

사후 평가 잘한 점 및 아쉬운 점

👍 잘한 점 (Key Success Factors)

✓ 물리 엔진 최적화

시뮬레이션 환경 내 안정적인 물리 연산 구현 성공 및 충돌 처리 최적화

✓ Phase 기반 로봇 제어

순차적 로직 적용으로 충돌 없는 안정적인 픽앤플레이스 (Pick & Place) 달성

✓ 시스템 통합 및 이해도 향상

세분화된 코드의 시스템 통합 성공 및 USD 파일 구조 / 스킨 활용 능력 증대

⚠️ 아쉬운 점 (Limitations)

! 초기 구조 이해의 어려움

ROS2 와 Isaac Sim 간 통신 및 전체 시스템 아키텍처 파악에 시간 소요

! 로봇암 동작 속도 한계

실시간 처리량 증대를 위한 로봇암 작동 속도 및 모션 플래닝 개선 필요

Project Review Matrix

Success Points

Physics · Control · Teamwork

Limitations

Speed · Data · Initial Setup

Future Plan

GPU Optimization · Multi-Agent

사후 평가 개선 및 확장 방향

🔧 기술적 개선점 (Improvements)

⌚ 로봇암 작동 속도 개선

현재 속도 제약을 해소하여 사이클 타임 단축 및 처리량 증대

🔍 미세 결함 디텍팅 고도화

스크래치 등 미세 결함 학습 강화로 CNN 모델 정밀도 향상

▣ 적용 분야 확장 (Scalability)

☁️ 유사 외형 부품 분류

볼트, 너트 등 크기와 형태가 유사한 산업용 부품 분류로 확장

🌐 산업 전반 응용

과일, 광물 등 등급 분류가 필수적인 식품 및 제조 산업 응용

기대 효과

Expected Benefits



다른 산업으로 확장 가능

- 결함 검출
- 품질 검사
- 제품 분류

24h



무인 자동화

- 중단 없는 연속 공정 수행
- 완전 무인 시스템 구축 가능

사후 평가 추가할 수 있는 내용

Phase: Future Roadmap

팀명 : F-2 삼성

🔧 기술적 개선점 (Improvements)

⌚ 로봇 암 정지 기능 추가

태스크 완료 후 정지

🔍 협동로봇 시스템 기능 추가

모바일 로봇을 이용해 박스 이송



사후 평가 소감 및 향후 계획

▶ 프로젝트 수행 소감



이론의 실무 적용 (Theory to Practice)

익숙하지 않은 툴 (Isaac Sim, ROS2 연동)을 활용하여 비대면 수업에서 학습한 개념을 실제 프로젝트로 구현



개발 전체 사이클 경험

단순 구현을 넘어 기획 → 구현 → 개선으로 이어지는 프로젝트 완주



문제 해결 능력 향상

실무형 프로젝트 수행을 통해 돌발 이슈에 대처하는 응용력과 논리적 문제 해결 능력 배양



팀원 (Members' Review)



이승민 Team Leader

“ 시스템 통합 (System Integration)의 중요성을 몸소 체험했습니다 . ”



이주학 Team Member

“ Isaac Sim 기반 디지털 트윈 환경 구축 노하우를 습득했습니다 . ”



최대혁 Team Member

“ 객관적 근거에 기반한 데이터 중심 사고 (Data-Driven)를 확립했습니다 . ”