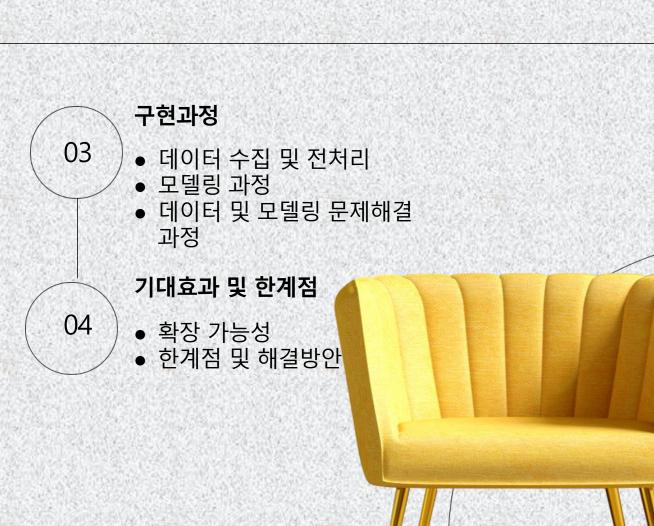


# 프로젝트 목차





### 프로젝트 제안 배경



#### 내 평생 153일 찾아 헤맨다.. 대체 무엇이기에?

등록 2012-03-28 오전 246:00 가 가





김민정 기자

N 기자구독

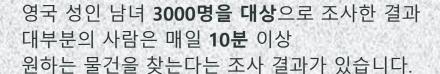


지금 열독 중



[이데일리 김민정 리포터] 대부분의 사람이 매일 잃어버린 물건을 찾기 위해 10분 이싱 3000명을 대상으로 조사한 결과, 대부분의 사람은 매일 10분 이상 자신이 원하는 물건 을 찾는데 시간을 소비하는 것으로 나타났다고 보도했다.







저희가 집중한 문제는 바쁜 현대인들의 일상적 고민인 물건 찾기 입니다. 피곤한 아침마다 핸드폰을 찾고, 중요한 약속 직전에 지갑을 찾는 당황스러운 순 간은 하루를 망치는 시작이 되곤 합니다.

특히 바쁜 현대인들에게 이런 일상적 스트레스를 해결하고자 프로젝트를 제안 하게 되었습니다.

### 프로젝트 제안 배경

자녀 둔 맞벌이 부부 필수템…新 홈캠 사용법 가정 내 홈 카메라 보유율 증가 추세 맞벌이 부부 사이서 '인기'



글로벌 스마트 홈 보안 카메라 시장은 2033년까지 561억 2천만 달러를 초과할 것입니다.

Spherical Insights & Consulting에서 발표한 연구 보고서에 따르면 **글로벌 스마트 홈 보안 카메라 시장** 규모는 2023년 98억 2천만 달러에서 2033년 561억 2천만 달러로 성장할 것으로 예상되며 2023-2033년 예측 기간 동안 19.04%의 CAGR로 성장할 것으로 예상됩니다.



출처 – 한경, Spherical Insights & Consulting

홈카메라의 보급이 증가하는 가운데, 단순 감시를 넘어 실생활에서 적극적으로 활용할 수 있는 AI 기반 물건 찾기 시스템이 필요합니다.

이는 고령층, 바쁜 현대인, 스마트홈 사용자에게 필수적인 기능이 될 것이며, 궁극적으로 시간 절약, 편의성 향상, 독립적인 생활 지원 등의 효과를 제공할 수 있습니다.

### 아이디어 소개

AI 기반 실시간 물건 인식 기존 CCTV의 단순 영상 저장을 넘어, 첨단 AI 기술로 실시간 객체 인식 및 추적을 실현하고 사용자의 소중한 물건을 24시간 모니터링합니다.

"내 핸드폰 어디 있어?"라는 간단한 음성 명령만으로 찾고자 하는 물건 의 위치를 즉시 확인할 수 있습니다.

편리한 음성 인터페이스



스마트한 위치 기록 시스템 물건이 이동할 때마다 AI가 자동으로 위치를 기록하고 업데이트합니다. 가구나 주변 물체를 기준점으로 활용 해 정확한 위치 정보를 저장하여합니다.

# **TEAM Member**



#### 김다솜

- 아이디어 기획
- 데이터 수집
- PPT 제작
- 모델학습



#### 변진섭

- 아이디어 기획
- 데이터 수집
- PPT 제작
- 모델학습



#### 이유진

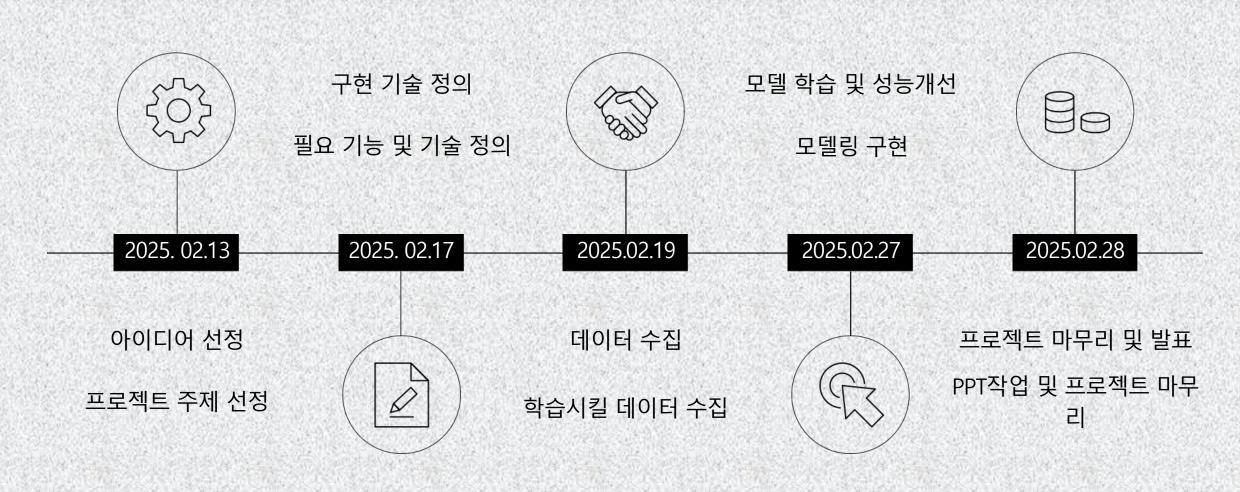
- 아이디어 기획
- 데이터 수집 라벨링
- 모델 학습 및 구축



#### 이창민

- 아이디어 기획
- 데이터 수집 라벨링
- 모델 학습 및 구축

# 프로젝트 일정



```
display: inline-blocks
                   CHAPTER
```

# 주요기능 및 기술스택

# 구현방식



타겟물체 음성 입력 실시간 물체 감지 및 추적 객체 인식(YOLOv8) / 물체 위치 추적(ByteTrack)

공간 관계 분석 참조 물체 기반 위치 파악 / 상대적 위치 관계 분석

위치 정보 저장 및 관리 물체 움직임 감지시 위치정보 업데이트



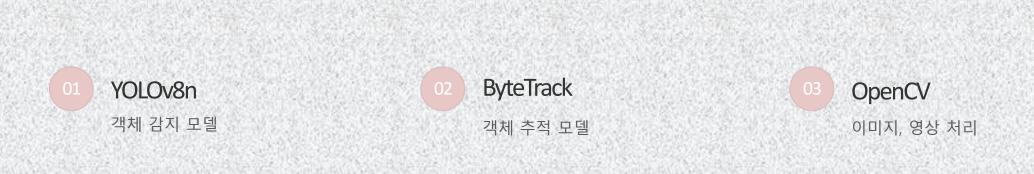
참조 물체 기반 위치 정보 제공

외출전

소지품 알림

음성 답변 제공

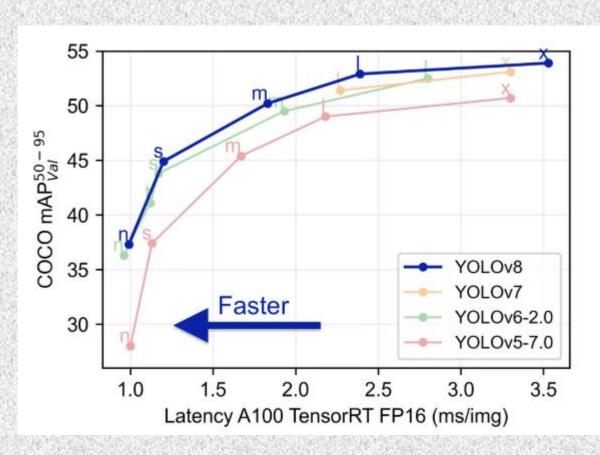
# 기술스택



# GHAPTED 3

구현 과정

### 모델선정

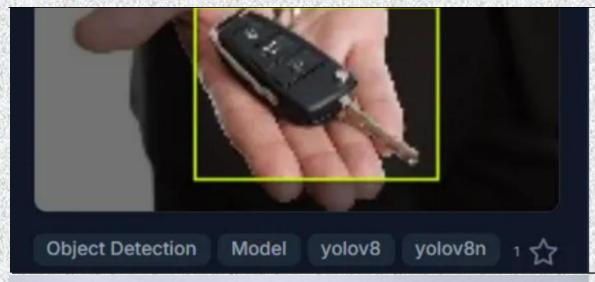


#### YOLOv8n

- 가장 가벼운 모델
- 높은 정확도를 제공, 실시간 객체 감지에 최적
- 모델 경량화, 정확도, 속도 간의 밸런스가 가 장 뛰어난 모델

출처 : Ultralytics

# 데이터셋 수집



#### Roboflow / coco dataset

- ☑ 수집데이터 약 60,000개
- ☑ train: 70% / valid:20% / test: 10%
- ☑ 타겟 객체(예: 차 키, 지갑) / 참조 객체(예: 책상, 소파)



#### 커스텀 데이터

- ☑ 직접 촬영한 스마트폰/지갑 사진
- ☑ 수집데이터 약 100개
- ☑ 촬영 배경: 사무실 책상, 바닥 등 일상적인 환경

# 데이터 전처리 과정 (Data Preprocessing)

```
train: ../train/images
val: ../valid/images
test: ../test/images

nc: 3
names: ['Chair', 'Sofa', 'Table']

roboflow:
  workspace: d1-kdezi
  project: furniture-ngpea-u63cc
  version: 1
  license: CC BY 4.0
  url: https://universe.roboflow.com/d1-kdezi/furniture-ngpea-u63cc/dataset/1
```

#### 라벨링 재구성

☑ 불필요한 Class 목록 제거

#### 라벨링 작업

- ☑ 수집된 데이터 통합
- ☑ 하나의 data.yaml 파일 생성
- ☑ 기존 label 값을 새로운 names 배열의 인덱스에 맞춰 변환
- ☑ 직촬영한 스마트폰/지갑 사진 등
- ☑ 촬영 배경: 사무실 책상, 바닥 등 일상적인 환경

# 데이터 전처리 과정 (Data Preprocessing)

#### 탐지 클래스 경량화

☑ 학습시간 및 클래스 탐지율 저하 문제로 자주 사용될 클래스에 집중하여 성능개선 후 클래스 확장



```
train: /home/alpaco/lcm/datas/dataset/train/images
val: /home/alpaco/lcm/datas/dataset/val/images
test: /home/alpaco/lcm/datas/dataset/test/images
nc: 6 # 총 클래스 수
names: ["person", "cell phone", "couch", "wallet", "desk", "drawer"]
```

# 데이터 통합 및 분할 (Data Integration & Splitting)

```
home/alpaco/findgo/dataset/
    train (70%)
          images
          labels
    valid (20%)
          images
          labels
    test (10%)
          images
          labels
    dataset.yaml
```

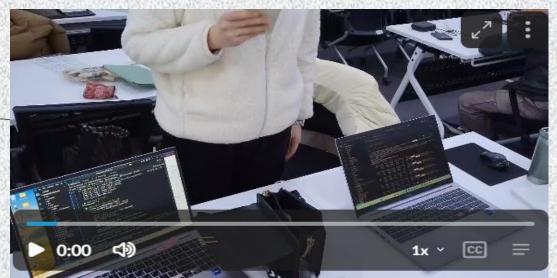
```
Train:
   "person": 3500
  "cell phone": 3500
  "couch": 3500
  "wallet": 3500
  "desk": 3500
  "drawer": 3500
Valid:
   "person": 1000
  "cell phone": 1000
  "couch": 1000
  "wallet": 1000
  "desk": 1000
   "drawer": 1000
Test:
   "person" : 500
  "cell phone": 500
  "couch" : 500
  "wallet" : 500
   "desk": 500
   "drawer" : 500
```

# 데이터셋 수집



#### 테스트 데이터

- ☑ 직접 촬영한 집/내부 영상 등
- ☑ 촬영 배경: 집 책상, 바닥 등 일상적인 환경



# 데이터 전처리 과정 (Data Preprocessing)

### 2. 라벨 데이터 검증( YOLOv8 형식)

데이터의 수량 파악 (데이터 불균형)

YOLOV8 .txt 형식 확인 (class\_id x\_center y\_center width height)

좌표값이 0 ~ 1 범위 내에 있는지 검토

빈 파일 및 빈 줄 여부 확인

이미지와 라벨 파일이 정상적으로 매칭되는지 검토

랜덤 샘플을 시각적으로 검토하여 바운딩 박스(BBox)가 올바르게 적용되었는

지 확인

# 데이터 전처리 과정 (Data Preprocessing)

### 3. 클래스 ID 재매핑

데이터 균형화

데이터셋 내 클래스 체계를 일관되게 정리

person
cell phone
couch
wallet
desk
drawer





# 학습 진행 및 평가 (Training & Evaluation)

### 성능 모니터링

Overfitting / Underfitting 여부 확인

Overfitting / Underfitting 여부 확인

Validation 및 Test 데이터셋 평가

# 학습 과정에서 성능 변화

### 첫 번째 학습 결과

person 클래스 성능이 현저히 낮음

mAP50: 0.0739

mAP50-95: 0.0454

#### [valid]

Ņ.													63
	Class	Images	Instances	Box(P	R	mAP50	mAP50-95): 1	00%	38	38 [00:	:06<00:00,	5.51it/s]	100
9	all	1205	2414	0.771	0.73	0.767	0.581						8
	person	200	200	0.211	0.07	0.0739	0.0454						- 8
8	cell phone	200	251	0.935	0.88	0.947	0.698						200
ġ	couch	200	220	0.745	0.627	0.701	0.499						100
8	wallet	208	229	0.906	0.97	0.967	0.733						2
	desk	197	249	0.9	0.876	0.939	0.731						20
8	drawer	200	1265	0.93	0.955	0.973	0.778						10

#### [test]

Class	Images	Instances	Box(P	R	mAP50	mAP50-95):	100%  38/38 [00:09<00:00, 4.02it/s]
all	605	1268	0.745	0.71	0.734	0.559	
person	100	100	0.235	0.0893	0.0782	0.0403	
cell phone	100	159	0.888	0.679	0.756	0.507	
couch	100	111	0.666	0.664	0.716	0.544	
wallet	105	113	0.92	0.991	0.988	0.797	
desk	100	134	0.824	0.858	0.884	0.68	
drawer	100	651	0.939	0.978	0.982	0.785	

### 학습 과정에서 성능 변화

### 첫 번째 학습 결과

"person" 클래스의 Precision-Recall 성능이 매우 낮음 (0.074)

모델이 "person" 클래스를 거의 탐지하지 못함

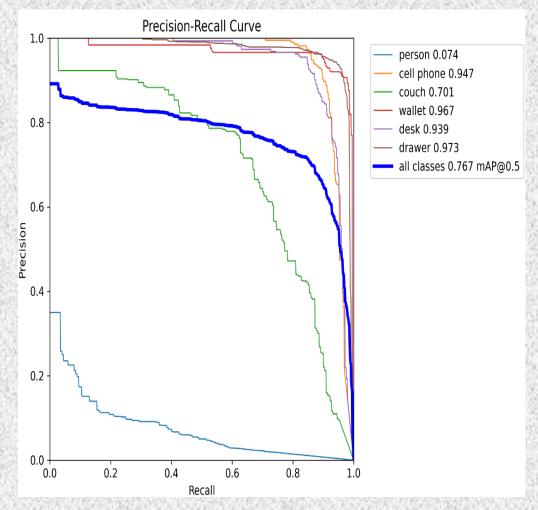
제대로 학습이 이루어지지않음

Bounding Box (BBOX) 검증 후,

작은 객체는 제대로 학습되지 않는다는 문제 확인.

작은 객체 대신 큰 객체가 포함된 데이터 수집

Bounding Box 크기가 충분한 새로운 데이터셋으로 재학습



test

### 학습 과정에서 성능 변화

# COCO Dataset에서 받은 데이터 중 BBox 좌표가 큰 데이터 기준으로 상위1000개 추가 후 새롭게 학습

#### [valid] Class Box(P mAP50-95): Images Instances 1205 person 200 200 0.211 0.07 0.0739 0.0454 cell phone 200 251 0.935 0.88 0.947 0.698 200 220 0.745 0.627 0.701 0.499 couch wallet 208 229 0.906 0.97 0.967 0.733 desk 0.731 197 249 0.9 0.876 0.939 drawer 200 1265 0.93 0.955 0.973 0.778

Class	Images	Instances	Box(P	R	mAP50	mAP50-95):
all	605	1268	0.745	0.71	0.734	0.559
person	100	100	0.235	0.0893	0.0782	0.0403
cell phone	100	159	0.888	0.679	0.756	0.507
couch	100	111	0.666	0.664	0.716	0.544
wallet	105	113	0.92	0.991	0.988	0.797
desk	100	134	0.824	0.858	0.884	0.68
drawer	100	651	0.939	0.978	0.982	0.785

Class	Images	Instances	Box(P	R	mAP50	mAP50-95
all	1205	2418	0.882	0.84	0.912	0.6
person	200	204	0.887	0.907	0.946	0.81
cell phone	200	251	0.921	0.873	0.932	0.68
couch	200	220	0.814	0.557	0.741	0.53
wallet	208	229	0.893	0.951	0.969	0.69
desk	197	249	0.863	0.831	0.929	0.67
drawer	200	1265	0.914	0.921	0.958	0.73
est]						
Class	Images	Instances	Box(P	R	mAP50	mAP50-9
all	605	1269	0.891	0.817	0.868	0.60
person	100	101	0.907	0.891	0.933	0.80

0.905

0.76

0.972

0.875

0.93

111

113

134

couch

wallet

100

105

100

0.656

0.586

0.828

0.942

0.735

0.677

0.991

0.904

0.969

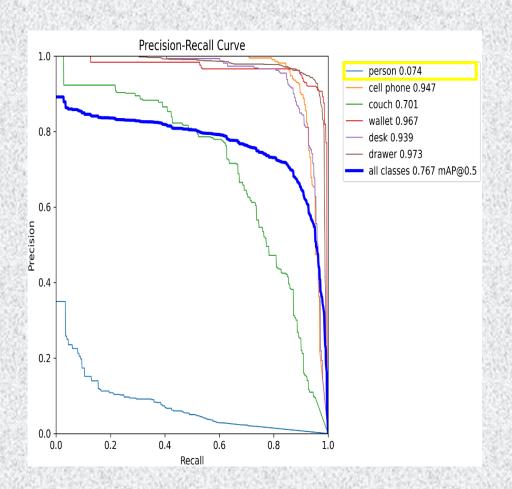
0.51

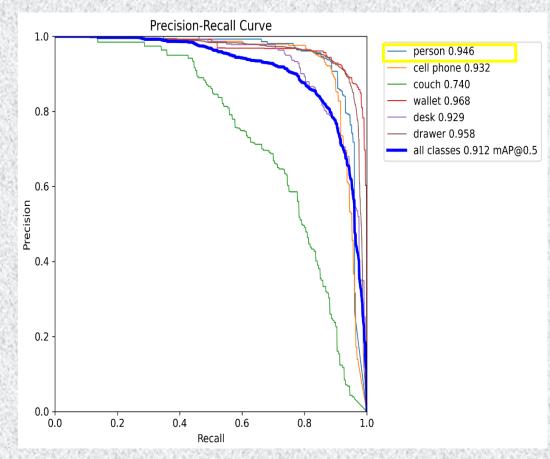
0.506

0.756

0.645 0.751

# 학습 과정에서 성능 변화





### 데이터 증강

Transform	Albumentations 2.0.2	AugLy 1.0.0	imgaug 0.4.0	Kornia 0.8.0	Torchvision 0.20.1
Perspective	1222 ± 16	-	908 ± 8	154 ± 3	147 ± 5
RandomResizedCrop	4521 ± 17	-	-	661 ± 16	837 ± 37
CoarseDropout	15760 ± 594	-	1190 ± 22	-	-
RandomBrightnessContrast	11254 ± 418	2108 ±	1076 ± 32	1127 ± 27	854 ± 13

실시간 객체 탐지 성능을 극대화하려면 Albumentations를 활용한 데이터 증강이 필수적

출처 : GitHub

#### Perspective

물건이 다양한 각도에서 보일 수 있도록 원근 변환 적용

RandomResizedCrop

물건이 다양한 크기와 위치에서 나타날 수 있도록 이동, 회전, 크

기 조정

RandomBrightnessContrast / RandomShadow

조명 변화 및 그림자에 강인한 모델을 학습하도록 대비 조절

CoarseDropout

물건이 부분적으로 가려져도 인식 가능하도록 일부 픽셀 제거

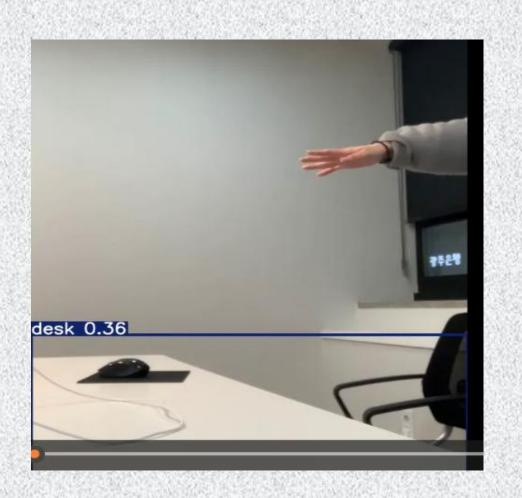








# 모델 테스트각 클래스별 데이터 1000개 vs 5000개 비교 실험



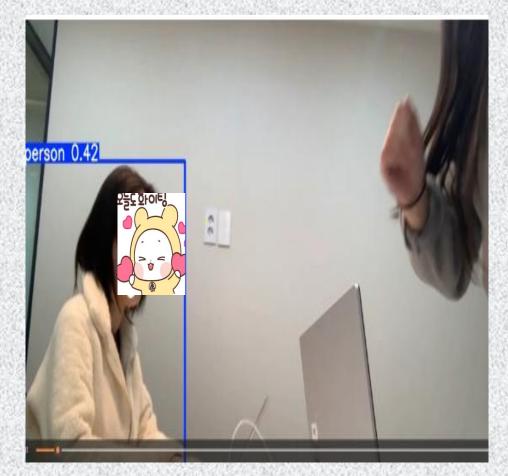




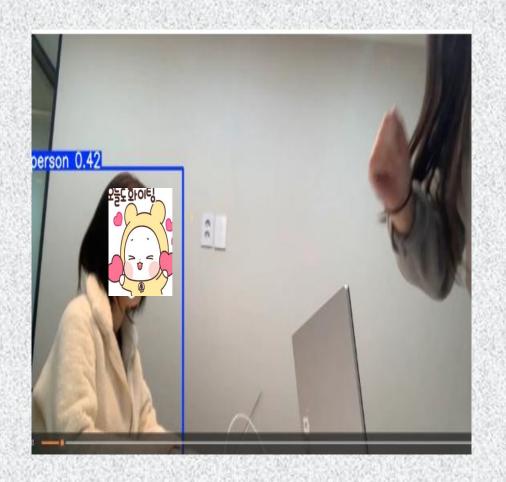
# 모델 테스트각 클래스별 데이터 1000개 vs 5000개 비교 실험







# 모델 테스트이미지 속 2개 이상의 객체 데이터 추가 학습







# 성능 개선

Problem 01

#### 문제 상황

- 물체가 연속적으로 움직일 때 간헐적으로만 감지됨
- 각도가 조금만 달라져도 객체 추적이 끊김
- 물체가 일시적으로 가려질 경우 객체 인식 실패
- 손이나 다른 물체에 의한 부분 가림에 매우 취약

Problem 02

#### 문제 원인

- YOLOv8은 각 프레임을 독립적으로 처리하여 연속성 부족
- 기존 추적 알고리즘은 신뢰도가 낮은 탐지 결과를 모두 버림
- 가려진 물체나 각도가 변한 물체는 낮은 신뢰도로 인해 무시됨

### 성능 개선

#### **ByteTrack: Multi-Object Tracking by**

Yifu Zhang<sup>1</sup>, Peize Sun<sup>2</sup>, Yi Jiang Zehuan Yuan<sup>3</sup>, Ping Luo<sup>2</sup>,

<sup>1</sup>Huazhong University of Science and Technology

#### **Abstract**

Multi-object tracking (MOT) aims at estimating bounding boxes and identities of objects in videos. Most methods obtain identities by associating detection boxes whose scores are higher than a threshold. The objects with low detection scores, e.g. occluded objects, are simply thrown away, which brings non-negligible true object missing and fragmented trajectories. To solve this problem, we present a simple, effective and generic association method, tracking by associating almost every detection box instead of only the high score ones. For the low score detection boxes, we utilize their similarities with tracklets to recover true ob-

# 해결 방안

80

77

75

70

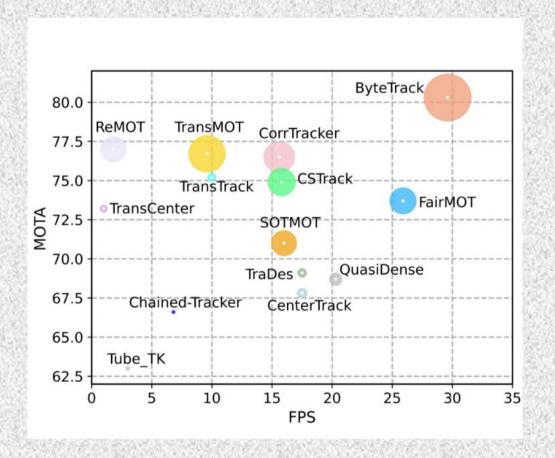
67

낮은 신뢰도를 활용한 객체추적기 도입

- ByteTrack 알고리즘으로 낮은 신뢰도의 탐지 결과도 활용
- 탐지 결과를 높은 신뢰도와 낮은 신뢰도로 분류하는 2단계 접근법 적용
- 1단계: 높은 신뢰도 탐지와 기존 추적 객체를 매칭
- 2단계: 미매칭 객체에 낮은 신뢰도 탐지 연결
- 가려지거나 각도가 변한 물체도 동일한 ID로 지속 추적 가능

출처: ByteTrack논문

## 성능 개선



# MOT17 데이터셋 기준 최 고 성능 달성

기존 알고리즘보다 우수한 성능

- ByteTrack: MOTA 80.3%, IDF1 77.3% 기록
- 기존 최고 성능 대비 약 5% 이상 향상된 수치

# 구현 방법

```
# 기본 주석 구행

try:

results = model.track(

source=frame,

persist=True, # 이전 추적 정보 유지

conf=conf_threshold, # 탐지 신뢰도 임계값

verbose=False
)
```

### YOLOv8의 track() 함수 활용

- 단순 탐지 대신 프레임 간 연결성을 유지하는 추적 알고리즘 적용
- persist=True 옵션으로 이전 추적 정보 유지

## 개선 결과

총 처리 시간: 14.23초

```
평균 FPS: 36.18
결과 저장 경로: /home/alpaco/kds/test/out
총 처리 프레임: 515/515
객체 탐지된 프레임: 372 (72.2%)
추적된 고유 객체 수: 3
클래스별 탐지 횟수: {'cell phone': 372, 'couch': 8}
총 처리 시간: 21.15초
```

클래스별 탐지 횟수: {'cell phone': 350}

#### 탐지율 증가

객체 탐지 성능이 개선되어, 기존 대비 탐지 프레임 수가 6.3% 증가 (350 → 372) 또한, 객체가 가려지는 상황에서도 연속 적인 프레임에서 일관된 ID를 유지하여 추적 안정성 향상. 이와 함께, 누락된 탐지가 감소하여 전 반적인 탐지 신뢰도가 개선됨..

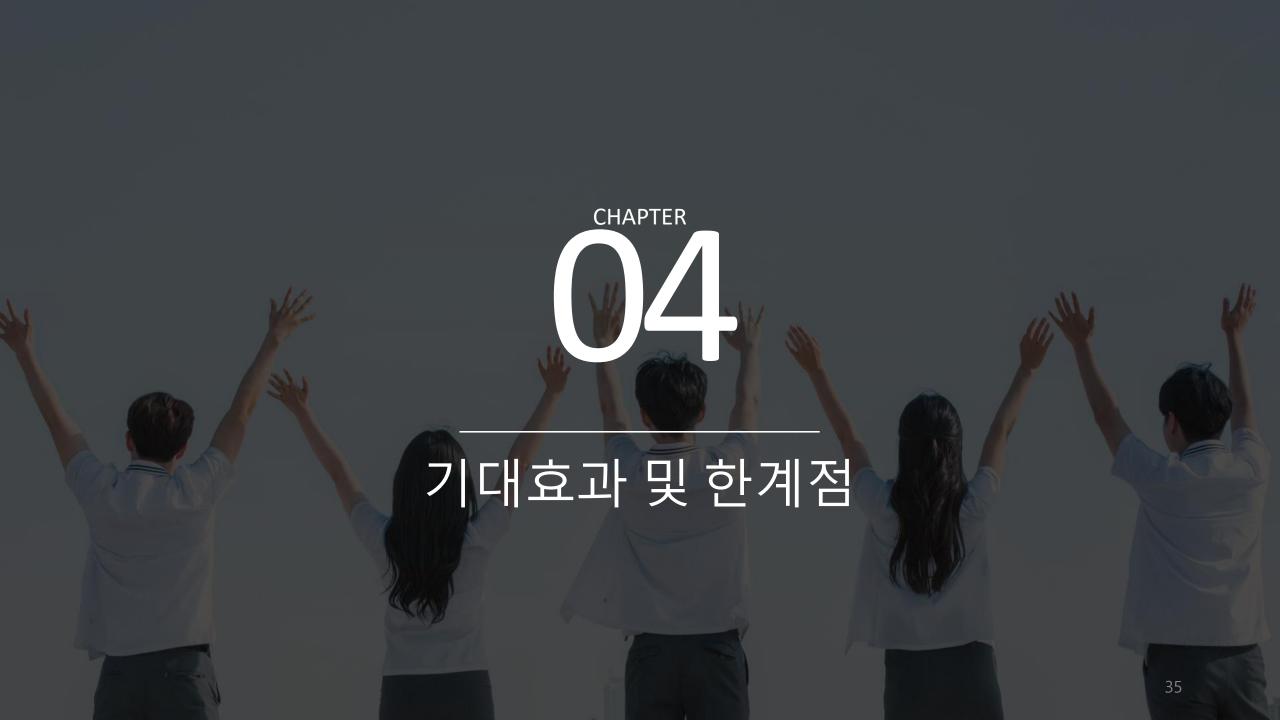
# 개선 결과

### 개선 전



### 개선 후





### 기대효과

- 이 홈카메라 활용 극대화
  - •기존 홈카메라는 주로 보안, 반려동물 및 자녀 모니 터링 용도로 사용됨.
  - •AI 기반 영상 분석 기술을 적용하면 가정 내 물건을 자동으로 인식하고 위치를 추적할 수 있음.
  - •스마트 홈 환경과 결합하여 사용자의 요청에 따라 즉각적인 물건 찾기 기능 제공 가능.

- 고령층 및 알츠하이머 환자의 일상 생활 보조
  - •기억력 감퇴로 인해 물건을 자주 잃어버리는 문제 해결 •"내 지갑 어디 있어?", "안경 어디에 뒀어?" 같은 질문에 AI 가 실시간으로 위치를 안내

- loT 및 스마트홈과의 연 계 가능성
  - •스마트 스피커(Alexa, Google Assistant), 스마트폰, 스마트 홈카메라와 연동 가능
  - •loT 센서 및 RFID 태그를 활용하여 물건을 실시간으로 추적하고 알림 제공

### 한계점

- □ 사생활 보호 문제
  - •집 안에 지속적으로 카메라가 작동하므로 프라이버시 문제 발생 가능
- 해결방안 로컬 데이터 저장 & 음성 명령 시 만 활성화

- 유사한 물건 구분 문제
  - •사용자가 찾는 물건이 아닌 비슷한 디자인의 핸드폰, 지갑을 잘못 인식할 가능성 있음
  - 해결방안 특징점 기반 추가 분석 모델 적용

- 사용자 경험(UX) 및 활 용 문제
  - •고령층 및 기술에 익숙하지 않은 사용자의 접근성 문제
  - 해결방안 고령층 및 기술 비숙련자를 위해 직관적인UI, 명확한 안내, 다양한 입력 방식, 접근성 강화로 사용 편의성을 높임

# Thank You