

AI 서비스 기획 · PM 포트폴리오

TryAngle.

좋은 사진의 이유를 분석해
원하는 사진을 찍도록 돕는 AI 촬영 가이드

문제를 정의하고 시장의 빈 곳을 찾아, AI 서비스로 풀어낸 과정을 보여 드립니다.

● 김현수

팀장 · 서비스 기획 · 데이터 분석 · AI 개발 · iOS 개발 · 2025.09 – 현재

수상 · 선정 — 중앙대 실전창업교육 F.A.S.T 대상(TryAngle) · 중앙대 AI · SW 융합
우수성과 우수상(패션 리뷰 사진 연구) · 한이음 드림업 우수 프로젝트(TryAngle)



목차

01 TryAngle 한눈에 보기

솔루션 한 문장 · 앱 화면 흐름 · 내가 한 9단계

02 서비스 · 데이터 흐름

서비스 흐름 · 데이터 흐름 · 안내 알고리즘 · 백엔드 · AI 모델

03 문제에서 서비스까지

문제 정의 · 시장 공백과 차별점 · 해결 설계 · AI 전달 · 근거와 검증 · 운영

04 데이터 구조 · 사용 기술

로그 한 줄의 구조 · 도구마다 직접 한 일

05 연구 · 역할

패션 리뷰 사진 연구(PoV) · 역할과 결과 · 다음 단계

이 텍스트가 증명하는 세 가지

01 문제 정의

'잘 찍었나' 판정 대신, 고른 사진과의 차이를 줄이는 문제로

02 공백 · 차별점

찍은 뒤 고치는 앱들 사이에서 '찍는 순간 한 문장 안내'

03 AI 서비스 전달

원인 순서 · 한 번에 하나 · 자동 촬영, 폰 안에서 실시간

TryAngle

01 TryAngle 한눈에 보기

솔루션 한 문장 · 앱 화면 흐름 · 내가 한 9단계

원하는 구도를 말로 설명하고 맞추기 어려워 재촬영이 반복되고, 거리 · 높이 · 배치는 찍은 뒤 보정으로 고치기 어렵습니다.

[SOLUTION]

고른 레퍼런스와 지금 화면의 차이를
온디바이스 AI가 실시간으로 재서
한 번에 하나씩 행동으로 안내하고
맞으면 자동 촬영하는 촬영 가이드 앱

레퍼런스 없이 AI가 구도 3안을 계산하는 엔진은 서버에서 돌고 있고, 폰용 모델로 옮기는 중입니다.

레퍼런스를 고르면, 같은 장면이 나올 때까지 한 문장씩 안내합니다

• 구조 탭 네 개(홈 · 레퍼런스 · 카메라 · 활동). 카메라 탭의 바로찍기는 레퍼런스 없이 찍고, 찍은 사진은 활동 탭에 쌓이며 서버로 올라갑니다.



1. 홈

레퍼런스 고르기

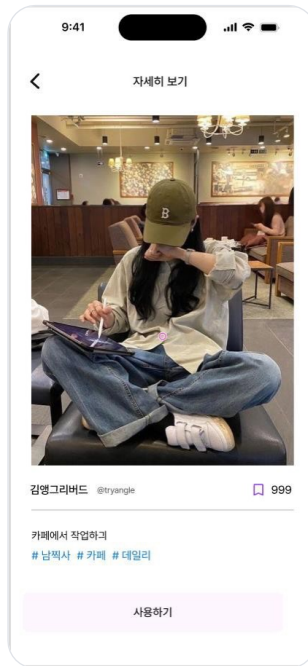
홈에서 바로 시작하거나
지난 분석 결과 보기



2. 레퍼런스

원하는 사진 찾기

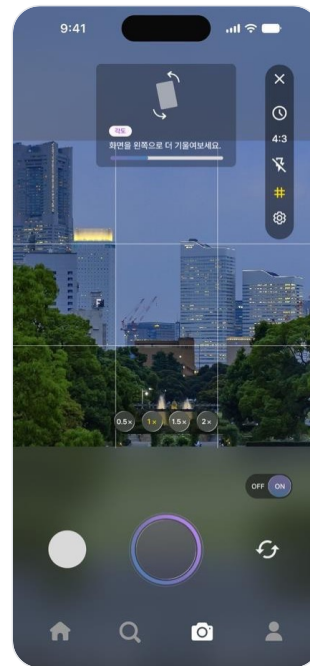
인물 · 스트리트 · 카페 ...
탭과 검색



3. 레퍼런스

사용하기

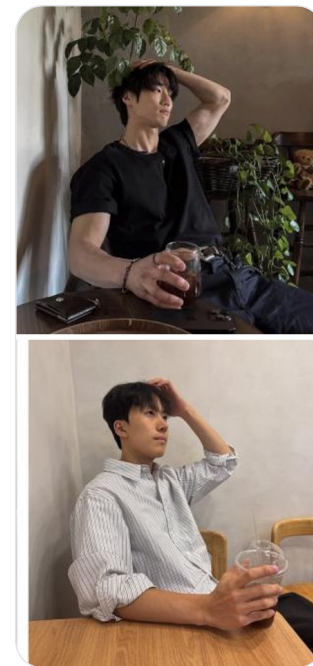
고른 사진이 목표가 됨



4. 카메라

한 문장 안내

고칠 것 하나씩, 맞으면
자동 촬영



5. 활동

결과 비교

위: 고른 사진 · 아래:
안내대로 찍은 결과

문제 정의부터 로그 개선까지, 9단계를 직접 연결했습니다

• **주황 테두리** 이 데크에서 자세히 다룬 단계 — 01 · 07 · 08 · 09

01 문제 정의

'잘 찍었나' 판정 → 고른 사진을 목표로

02 수집 · 라벨링

인물 사진 1만여 장 · 점수 3축 · 속성 9종 · 사람 검수

03 특징 · 탐색 분석

관절 · 카메라 · 3D 몸 · 임베딩 / UMAP · kNN · Cohen's d · EBM

04 분류 · 해석

장면 × 자세별 목표 구간 · 샷 범위 재현 확인

05 학습 전처리

잘라낸 화면 증강 · 묶음 단위 분할 · 제외 기준

06 모델 학습 · 경량화

교사 모델 4종 → 폰용 1개(지식 증류) · Core ML

07 사용자 검증

160문항 설문 · 평가자 간 일치도

08 앱 구현 · 베타

한 번에 하나 안내 · 약 26fps · TestFlight · 전시 체험

09 로그 재현 · 개선

1초 단위 로그 → 재생 → 원인 → 수정 → 재확인

TryAngle

02 서비스 · 데이터 흐름

서비스 흐름 · 데이터 흐름 · 안내 알고리즘 · 백엔드 · AI 모델

서비스 흐름 — 앱 · 서버 · 연구가 이어지는 한 바퀴

· 사용자는 레퍼런스를 고르고 안내대로 찍기만 하면 되고, 뒤에서는 서버와 연구가 기준과 모델을 계속 고칩니다.

1 레퍼런스 고르기

따라 찍을 사진을 고릅니다. 사진마다 거리 · 기울기 · 인물 크기 목표값이 미리 재어져 있습니다.

2 폰 안에서 분석 · 안내

카메라 화면을 폰 안에서 분석해 목표와의 차이를 재고, 먼저 고칠 것 하나를 한 문장으로 말합니다.

3 자동 촬영 · 결과

차이가 허용 범위 안에 머물면 자동으로 찍고, 활동 탭에 레퍼런스와 결과를 나란히 남깁니다.

4 기록 · 운영

분석값 · 안내 상태를 1초씩 묶어 서버에 남기고, 관리자 화면에서 재생해 막힌 곳을 찾습니다.

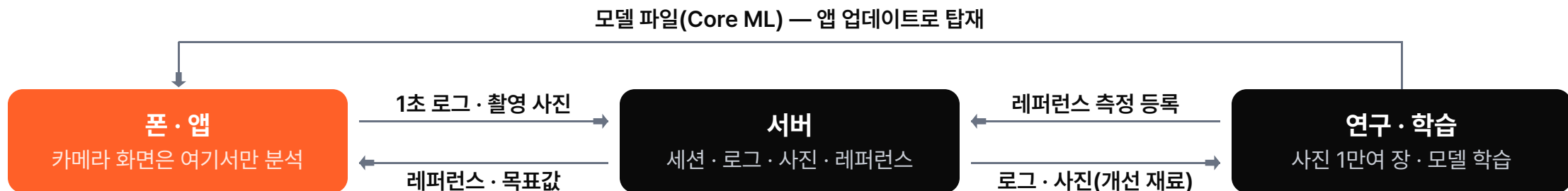
5 개선 · 연구

로그와 사진 1만여 장 분석으로 기준 · 문구를 고치고, 구도 추천 엔진을 폰용 모델로 옮깁니다.



데이터 흐름 — 무엇이 어디서 만들어져 어디에 쓰이나

• 카메라 화면은 폰 안에서만 쓰고, 서버에는 1초 묶음 수치와 찍은 사진만 올라갑니다. 연구 데이터는 기준 · 모델 · 레퍼런스가 되어 앱으로 돌아옵니다.



데이터	만들어지는 곳	가는 곳 · 저장	쓰임
카메라 화면	폰 카메라 · 30fps	폰 안에서만 — 밖으로 보내지 않음	사람 검출 · 관절 133점 · 이미지 모델 분석
분석값 · 안내 로그	앱 — 매 프레임 기록, 1초씩 묶음	서버 DB의 초 단위 표	세션 재생 · 막힌 단계 찾기 · 기준 개선
촬영 사진	앱 — 촬영 순간, 촬영 정보와 함께	S3 + 스냅 표 · 폰 앨범	활동 탭 결과 비교 · 운영 확인
레퍼런스 · 측정 문서	연구 — 사진마다 거리 · 기울기 · 샷 측정	서버 레퍼런스 표 → 앱 갤러리	촬영 목표값 · 레퍼런스 추천
라벨 표 10,709행	연구 — VLM 채점 + 사람 검수 + 3D 측정	분석 표(열 55개)	장면 × 자세 목표 구간 · 구도 추천 모델 학습
모델 파일	학습 서버 — 지식 증류	Core ML로 앱에 탑재	폰 안 실시간 추론

원칙 **원본 화면은 기기 밖으로 보내지 않음** **로그는 수치만 · 같은 초는 한 번만 저장** **레퍼런스는 측정값과 함께 등록**

안내 알고리즘 — 한 프레임이 한 문장이 되기까지

• 앞을 고치면 뒤 값도 바뀌므로, 원인에 가까운 것부터 하나만 말하고 흔들리지 않게 걸러서 보여 줍니다.

1 재기

매 프레임 관절 · 기울기 · 거리 · 인물 크기를 재서, 레퍼런스 목표값과 비교합니다.

2 원인 순서로 하나 고르기

거리 → 줌 → 높이 → 기울기 → 위치 → 방향 → 수평 순으로 보고 처음 어긋난 하나만 고릅니다.

3 흔들림 거르기

고른 축은 8프레임 이어져야 확정하고, 문장은 새 지시가 0.7초 넘게 이어질 때만 바꿉니다.

4 단계 확인

사람 인식부터 촬영까지 9단계를 앞에서부터 통과하고, 앞 단계가 틀어지면 0.5초 뒤 되돌아갑니다.

5 촬영

모든 단계가 5프레임 연속 통과하면 촬영 가능, 0.5초 유지되면 자동 촬영합니다(레퍼런스 모드).

카메라 30fps

밀린 프레임은 버리고 최신 화면만

1 재기

관절 133점 · 기울기 · 거리 · 인물 크기

2 원인 순서로 하나 고르기

거리 → 줌 → 높이 → 기울기 → 위치 → ...

3 흔들림 거르기

축 8프레임 유지 · 문장은 0.7초 넘게 이어질 때 교체

4 9단계 확인

사람 인식부터 촬영까지 앞 단계부터 통과

5 촬영

5프레임 연속 통과 → 0.5초 유지 시 자동 촬영

함께 도는 것

1초 묶음 로그 → 서버

발열 단계에 따라 분석
간격 조절

사람이 15초 없으면
무거운 분석 쉼

백엔드 — 촬영 한 번이 서버에 남는 과정

• 앱은 세션을 열고 1초마다 수치 로그를 보내며, 찍은 사진을 올린 뒤 세션을 닫습니다.

1 세션 시작

촬영 화면에 들어가면 레퍼런스 · 바로찍기 모드와 기기 정보로 세션을 엽니다.

2 1초마다 로그

분석값 · 안내 단계 · 화면 문장을 1초씩 묶어 보내고, 서버는 (세션, 초)마다 한 번만 저장합니다.

3 촬영 사진 등록

찍은 사진은 S3에 올리고, 레퍼런스 촬영이면 레퍼런스 번호와 함께 스냅으로 등록합니다.

4 세션 종료 · 조회

세션을 닫으면 저장된 초 수를 돌려받고, 관리자 화면의 세션 목록 · 재생 · 통계에서 봅니다.

5 직접 한 부분

1초 묶음 로그 구조를 설계하고, 관리자 화면에 세션 재생 분석 · 성능 진단을 더했습니다.



AI — 폰 안의 실시간 모델과 서버의 구도 추천 엔진

• 지금 앱은 폰용 모델로 실시간 안내하고, 레퍼런스 없이 구도를 고르는 엔진은 서버에서 돌며 폰으로 옮겨 갑니다.

1 폰 안 · 사람과 관절

사람을 찾아(Apple Vision) 그 영역만 잘라 관절 133점을 재고, 발열에 따라 간격을 조절합니다.

2 폰 안 · 이미지 모델

샷 · 몸 방향 · 거울 셀카 · 옷 · 점수를 한 번에 내는 모델 하나를 2프레임마다 돌립니다.

3 모델 만들기 · 지식 종류

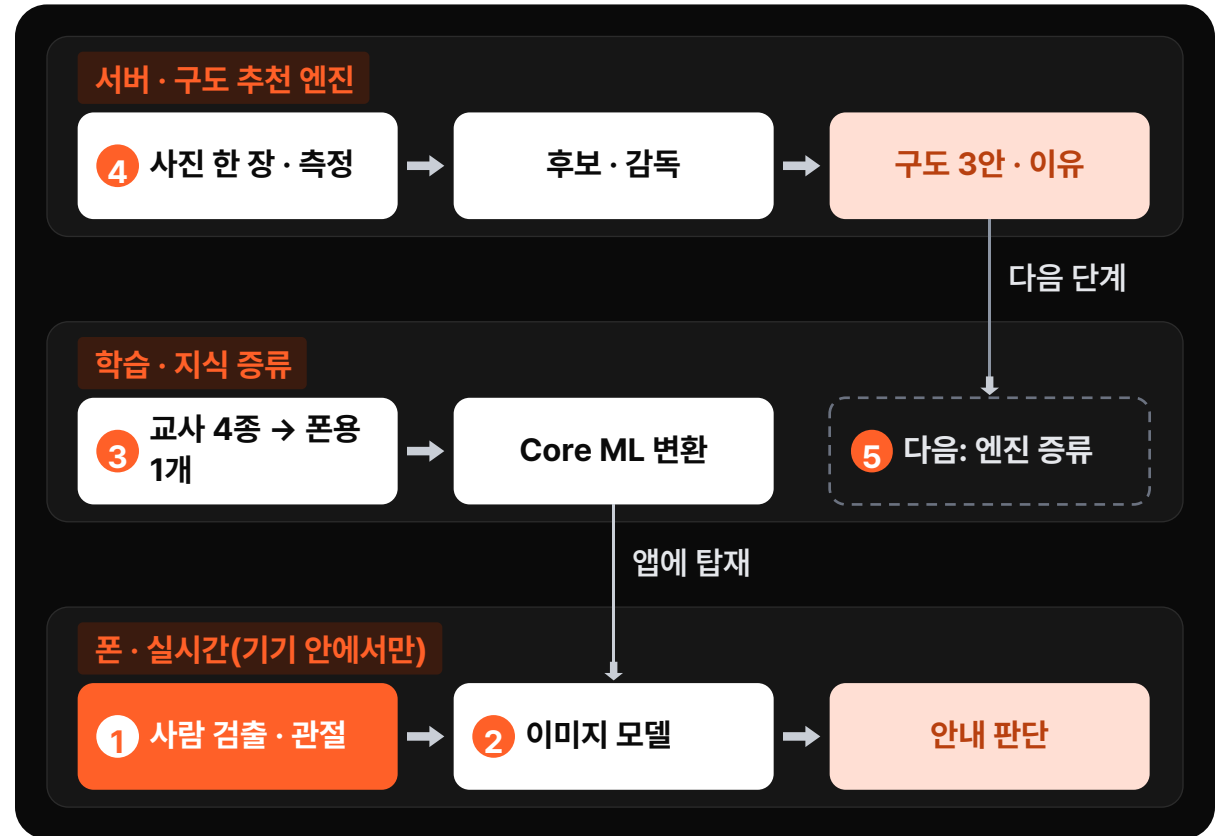
관절 · 기하 · 3D 몸 · 시각 특징을 내는 교사 4종의 답을 폰용 모델 하나가 배우게 했습니다.

4 서버 · 구도 추천 엔진

사진 한 장에서 몸 3D · 초점 · 장면을 재고, 서는 자리와 카메라 조작이 다른 구도 3안을 냅니다.

5 다음 · 폰으로 옮기기

엔진이 쓰는 값만 내는 폰용 단일 모델로 증류해, 레퍼런스 없는 바로찍기에 붙입니다.



TryAngle

03 문제에서 서비스까지

문제 정의 · 시장 공백과 차별점 · 해결 설계 · AI 전달 · 근거와 검증 · 운영

'잘 찍었는지' 판정하지 않고, 사용자가 고른 사진을 목표로 삼았습니다

- **문제** 좋은 사진의 기준은 사람과 목적마다 다릅니다. 점수 하나로 판정하면 사용자가 원하는 사진과 어긋납니다.
- 근거: 1만여 장에서 미학 점수와 유용성(옷 정보가 잘 보이는 정도) 점수의 상관 -0.08 — 한 점수로 묶을 수 없음

해결

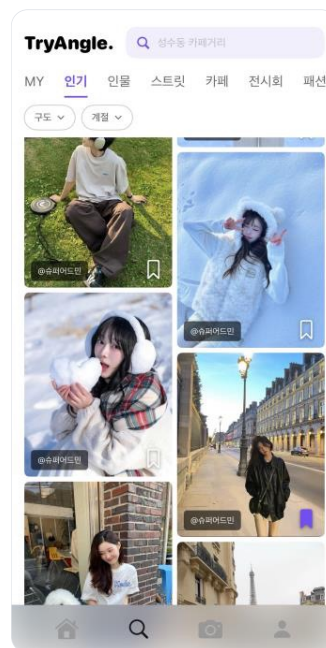
문제를 '사용자가 고른 레퍼런스와 지금 화면의 차이 줄이기'로 다시 정의했습니다.
앱은 취향을 판정하지 않고 거리 · 높이 · 방향 · 위치의 차이만 재서, 한 번에 하나씩 안내합니다.

사용 기술

서비스 흐름 설계

Figma · 화면정의서

요구사항 정리



레퍼런스 고르기
따라 찍고 싶은 사진



지금 화면 분석
관절 · 기울기 · 위치 측정



차이 하나 안내
가장 먼저 고칠 것 하나



촬영
목표에 가까워지면

기존 앱이 비워 둔 곳 — 찍는 순간의 행동 안내

• 상용 카메라 앱은 촬영 → 필터 → 보정 흐름은 갖췄지만 구도를 잡는 단계가 비어 있고, 찍은 뒤 보정으로는 거리 · 높이 · 배치를 고칠 수 없습니다.

항목	오버레이형(SOVS)	구도 제안형(Frame)	후보정형(Snow · Meitu)	TryAngle
시점	촬영 중(실시간 아님)	촬영 중(실시간 아님)	촬영 후	촬영 중(실시간)
피드백	레퍼런스를 겹쳐 표시	일반 구도 규칙	필터 · 보정 · 스티커	사용자 맞춤 행동 지시
안내 기준	예시와 직접 비교	일반 구도 규칙	촬영된 사진	목표 사진과의 차이
사용자 조작	겹친 화면에 맞춤	제안 구도에 맞춤	필터 · 보정 적용	한 문장 안내 수행
주요 역할	구도 · 포즈 참고	구도 선택 보조	촬영 결과 보정	안내 후 자동 촬영

비어 있던 곳 **촬영 중 실시간** **내 체형에 맞춤 목표** **바로 할 수 있는 행동 한 가지**

빈 곳을 네 가지 설계로 풀었습니다

• 레퍼런스를 목표로 삼되, 사람마다 다른 체형과 흔들리는 측정값, 기다리는 시간을 설계로 흡수했습니다.

1 내 체형에 맞춘 목표

레퍼런스와 지금 화면의 상대 신체 비율을 비교해, 키 · 체형이 달라도 같은 구도가 되게 안내합니다.

2 한 번에 하나, 해결될 때까지

먼저 고칠 요소 하나만 고르고 해결될 때까지 같은 지시를 유지해, 안내가 번갈아 바뀌지 않게 했습니다.

3 폰 안에서 바로

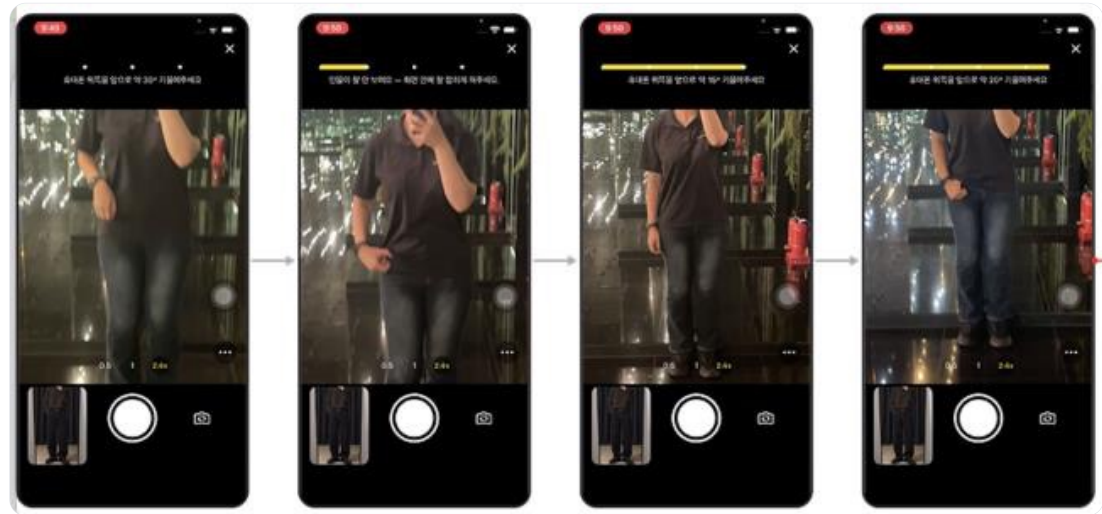
촬영 화면을 서버로 보내지 않고 기기 안에서 분석해, 통신을 기다리지 않고 영상도 밖으로 나가지 않습니다.

4 맞으면 자동 촬영

촬영 가능 상태가 0.5초 이어지면 자동으로 찍고, 결과를 레퍼런스와 나란히 보여 줍니다.

5 다음: 레퍼런스 없이

레퍼런스를 고르지 않아도 서는 자리와 카메라 높이를 제안하는 구도 추천 AI를 폰으로 옮기는 중입니다.



안내가 한 문장씩 바뀌어 가는 실제 촬영 화면

0.5초

촬영 가능이 이어지면
자동 촬영

8프레임

고른 지시가 이어져야
확정

기기 안

촬영 영상은 밖으로
보내지 않음

원인부터 순서대로, 한 번에 하나만 말하도록 안내 정책을 설계했습니다

- 문제 거리 · 높이 · 기울기 · 위치를 한꺼번에 알려 주면 무엇부터 할지 모르고, 한 가지를 고치면 다른 값도 따라 바뀝니다.

해결

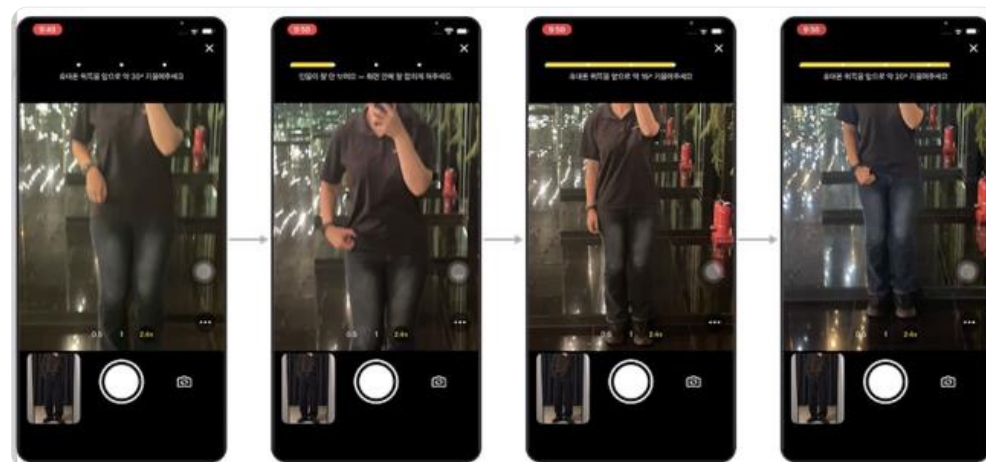
앞 단계가 뒤를 바꾸는 순서(거리 → 줌 → 높이 → 기울기 → 위치 → 방향 → 수평)를 정해 첫 번째로 어긋난 것 하나만 말하게 했습니다. 측정값이 흔들려도 문장이 깜빡이지 않도록, 새 지시가 8프레임 연속 골라지고 0.7초 넘게 이어질 때만 바꿉니다.

사용 기술

안내 정책 설계

상태 전환 규칙

Swift



거리 → 줌 → 높이 → 기울기 → 위치 → 방향 → 수평

앞 단계가 뒤를 바꾸는 순서 — 첫 번째로 어긋난 것 하나만 말한다

독립변수부터 맞추고, 따라 정해지는 것은 뒤로 미뤘습니다

• 앞 단계를 고치면 뒤 단계 값이 따라 바뀌므로, 원인이 되는 값부터 맞추도록 게이트 순서를 정했습니다.

게이트 9단계

사람 인식 → 거울 확인 → 화면비 → 사진 종류 → **프레이밍** → 카메라 조정 → 위치 → 자세 → 촬영

1. 먼저 — 독립변수

거리 · 줌(화각)

따라 정해지는 것(종속변수)

인물 크기(프레이밍) · 배경 압축감

크기는 줌으로도 맞출 수 있지만, 배경 압축감은 거리가 정해서 둘 다 맞춥니다.



2. 그다음 — 카메라 각도

높이 · 기울기(틸트)

바뀌는 것

위아래 시점 · 화면 속 인물 높이

인물 크기는 거의 그대로 두고 시점과 높이만 바뀌어서, 크기가 정해진 다음 차례입니다.



3. 마지막 — 미세 조정

위치 · 방향 · 수평

다듬는 것

머리 위 여백 · 좌우 균형 · 몸 방향

위치는 앞 단계의 결과라 앞이 맞기 전엔 말하지 않고, 방향 · 수평은 차례만 뒤로 둡니다.

처음 문제

축마다 따로 판정하니 거리와 위치, 기울기와 위치처럼 묶인 축끼리 안내가 4건 부딪혔습니다.

바꾼 규칙

처음 어긋난 하나만 말하고, 샷 크기처럼 결과로 정해지는 값은 따로 말하지 않습니다.

미룬 이유 기록

미룬 축은 '고치면 따라올 값'과 '차례만 뒤인 값'으로 나눠 로그에 남깁니다.

분석 모델을 폰 안에서 돌려, 촬영 중 실시간으로 안내합니다

- 문제 사진을 서버로 보내 분석하면 안내가 늦어 촬영 흐름이 끊깁니다.

해결

사람 검출 · 관절 133점 · 6가지를 내는 이미지 모델을 모두 폰 안에서 돌리고, 발열 단계에 따라 분석 간격을 1~10프레임으로 조절했습니다. TestFlight로 실기기 테스트를 반복하고, 졸업전시에서 3일간 약 200명이 직접 써 보았습니다.

사용 기술

Swift · Core ML

Apple Vision

TestFlight 배포



왼쪽 레퍼런스를 보고, 안내대로 찍은 결과(오른쪽)

약 26fps

실기기 중앙값 ·
기기와 발열에 따라
15~33

약 79MB

폰에 담은 모델 합계

489건

실기기 촬영 세션 기록

데이터와 사용자 검증으로 기능을 정했습니다

• 감으로 정하지 않고, 분석 · 설문 · 로그에서 나온 근거를 결정으로 바꾸고 서비스에 반영했습니다.

결정	근거	서비스에 반영
점수로 판정하지 않는다	미학 ↔ 유용성 점수 상관 -0.08 — 좋은 기준이 목적마다 다름	고른 레퍼런스와의 차이를 줄이는 안내
목표는 장면 × 자세마다 따로	설명형 모델(EBM)에서 장면이 점수 기여도 1위	16칸 목표 구간(인물 크기 · 여백 · 각도)
한 번에 하나만 말한다	앞 단계를 고치면 뒤 값도 바뀜(거리 → 크기 · 위치)	원인 순서 안내 · 흔들림 방지 규칙
크기가 다른 안을 함께 둔다	설문에서 두 사람 모두 '더 넓게'를 고름(80% · 77%)	구도 추천 3안의 구성
위치 규칙은 늘리지 않는다	두 사람의 선택 일치 53%(우연이면 50%)	규칙 대신 사용자가 고르게 둬م
관절이 불확실하면 버린다	로그 재생: 사람이 떠난 뒤 관절 신뢰도 중앙값 0.11	믿을 만한 관절 6개 미만이면 결과를 버림

160문항 설문으로, 두 사람이 함께 고른 것과 사람마다 갈리는 것을 나눴습니다

- **문제** 데이터 분석은 '많이 쓰는 방식'을 알려 줄 뿐, 사용자가 실제로 더 좋아하는지는 알려 주지 않습니다.

확인 → 적용

크기 · 위치는 같은 사진을 잘라 만든 두 화면 비교, 배경 · 자세는 사진 한 장 질문으로 구성된 160문항을 두 명에게 진행했습니다. 두 사람 모두 '더 넓게'를 골랐지만(80% · 77%) 점수 높은 사진은 더 가까이 찍혀 있어, 구도 추천(사진 한 장 → 구도 3안)에 크기가 다른 안을 함께 두었습니다. 위치는 일치가 53%로 갈려 규칙을 늘리지 않았습니다.

사용 기술

160문항 설문 설계

평가자 간 일치도

결과 → 규칙 반영

항목	결과	결정
크기	'더 넓게' 담은 화면 선택 80% · 77%	크기가 다른 안을 3안에 함께 둠
인물 위치	두 사람 일치 53% — 취향이 갈림	지금 배치 규칙 유지, 더 늘리지 않음
배경 규칙	두 사람이 함께 거슬린다고 고른 사진 0/30장	서비스에 올리지 않음
자세 아이디어	나아짐 - 나빠짐 비율 +0.40	추천 결과에 계속 표시

크기 58 · 위치 47 = 두 화면 비교, 배경 30 · 자세 25 = 한 장 질문 · 평가자 2명 · 3안 = 구도 추천 서비스의 구도 3안

1초 단위 로그를 재생해, 현장에서 본 이상 동작의 원인을 찾고 고쳤습니다

- **문제** 현장에서 '안내가 이상했다'는 말만으로는, 모델이 잘못 본 것인지 규칙이 잘못된 것인지 알 수 없습니다.

해결

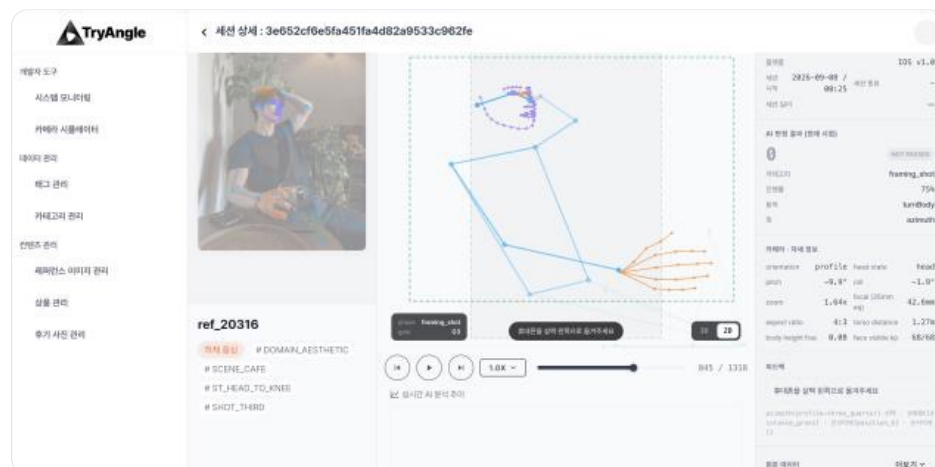
분석값 · 안내 단계 · 화면에 뜬 문장을 1초 묶음으로 남기는 로그 구조를 설계하고, 관리자 화면의 세션 재생에 분석 · 진단 기능을 더했습니다. 재생해 보니 사람이 화면을 떠난 뒤에도 관절을 지어내고 있어(신뢰도 중앙값 0.11), 믿을 만한 관절이 6개 미만이면 결과를 버리고 이런 프레임이 10개 이어지면 검출을 새로 시작하게 고쳤습니다.

사용 기술

로그 구조 설계(JSON)

재생 분석 · 성능 진단 추가

재현 검증



총 416개

세션 ID	사용자명	이미지 ID	시작일시	종료일시	디바이스	세션 상태	총 스냅샷 수	최대 현재 시간	주요 피드백 메시지
52538	fctef	1611	2026-07-03 / 2308	2026-07-03 / 2308	ios / 1.0	정상종료	7	18.3초	피사체를 인식할 수 없습니다. 화면 중앙에...
52544		1544	2026-07-03 / 2308	2026-07-03 / 2309	ios / 1.0	자동종료	2	1.0초	피사체를 인식할 수 없습니다. 화면 중앙에...
82592		1508	2026-06-28 / 1821	2026-06-28 / 1822	ios / 1.0	자동종료	5	3.8초	피사체를 인식할 수 없습니다. 화면 중앙에...
b2a6b		1510	2026-06-28 / 1821	2026-06-28 / 1822	ios / 1.0	자동종료	6	4.8초	피사체를 인식할 수 없습니다. 화면 중앙에...
30f84		1518	2026-06-28 / 1821	2026-06-28 / 1822	ios / 1.0	자동종료	4	2.8초	피사체를 인식할 수 없습니다. 화면 중앙에...

세션 목록(아래)에서 골라 스켈레톤 · 안내 문장을 초 단위로 재생(위)

TryAngle

04 데이터 구조 · 사용 기술

로그 한 줄의 구조 · 도구마다 직접 한 일

로그 한 줄의 구조를 먼저 정하고, 그 값으로 무엇을 볼지 연결했습니다

항목	뜻	예시	이 값으로 보는 것
phase · pidx	지금 흐름 단계(사람 인식 ~ 촬영, 0~8)	CAMERA_ADJUST · 5	어느 단계에서 오래 머무나
feedback	화면에 실제로 뜬 문장	휴대폰 위쪽을 뒤로 기울여주세요	사용자가 본 안내를 그대로 재생
axis · action	지금 고칠 축과 행동(거리 · 줌 · 높이 ...)	pitch · tiltBack	어떤 조정을 가장 많이 요구하나
stuckSec	지금 단계 경과 시간(초)	3.2	막히는 구간 찾기
canCapture	촬영 가능 여부	false	촬영 가능까지 도달했나
torsoDistM · pitchDeg	추정 거리 · 기울기	2.37 · -16.1	목표와의 차이가 줄어드나
perf	카메라 · 분석 fps, 발열	30.0 · 15.2 · fair	기기별 속도 · 발열 문제
kp	관절 133점(압축 문자열)	16140d32...	관리자 화면 스켈레톤 재생

설계 원칙

1초 묶음 전송 — 재생 단위와 서버 부하를 맞춤

화면 문장을 그대로 저장 — 사용자가 본 것이 분석 단위

로그에는 사진 없이 스켈레톤만

기획부터 운영까지, 도구마다 직접 한 일

기획 · 협업

Figma · 화면정의서로 사용자 흐름과 기능 요구사항 정리, 개발 · 디자인 간 구현 범위 조율

문제 정의 · 시장 분석

유사 서비스를 시점 · 피드백 · 안내 기준으로 비교해 비어 있는 단계를 정의

근거 기반 결정

pandas로 분석 결과를 직접 확인하고, 160문항 설문을 설계해 사람끼리 일치도로 해석

AI 서비스 설계

원인 순서 · 한 번에 하나 안내 정책, 흔들림 방지와 자동 촬영 조건 설계

로그 · 운영

1초 묶음 로그 구조 설계, 관리자 화면에 재생 분석 · 성능 진단 추가

구현 이해

Swift · Core ML로 실시간 안내를 직접 구현, TestFlight로 실기기 배포

사업화 문서

예비창업패키지 · 창업 공모전 사업계획서 작성, 발표(실전창업교육 대상)

AI 도구

Claude Code로 구현 속도를 높이고, 결과는 로그 재생 · 고정 사진 검사로 확인

TryAngle

05 연구 · 역할

패션 리뷰 사진 연구(PoV) · 역할과 결과 · 다음 단계

패션 리뷰 사진 연구 — 좋은 사진의 기준은 목적마다 달랐습니다

- **질문** 패션 리뷰 사진의 '좋음'을 점수 하나로 보면, 구매에 도움 되는 사진과 보기 좋은 사진이 섞입니다.

확인 → 적용

리뷰 유용성과 미학을 따로 라벨링하고 56개 특징을 비교했습니다. 출처를 섞으면 두 점수 상관이 -0.313 이지만 같은 쇼핑몰(무신사) 안에서는 0.010 으로, 출처가 만든 착시였습니다. 같은 특징이 두 점수에 반대로 작용해(EBM) 목적별 촬영 가이드 정책을 제안했습니다.

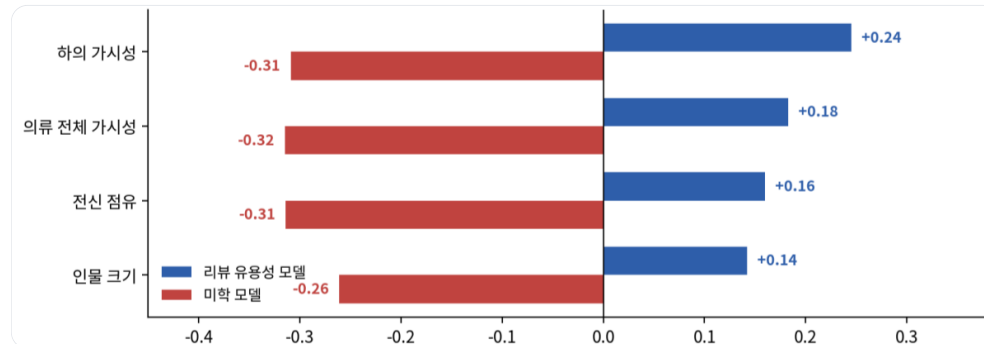
사용 기술

연구 설계

Cohen's d · EBM

출처별 층화 분석

2인 팀 팀장 · 2026.03 - 08 · 중앙대 AI · SW 융합 우수성과 발표대회 우수상



EBM 기여도 — 같은 특징이 유용성(파랑)과 미학(빨강)에 반대로 작용

-0.313

출처를 섞은 전체 상관

0.010

같은 쇼핑몰(무신사)
안 상관

.446 · .218

설명력 R^2 미학 ·
유용성

문제를 정의하고, 빈 곳을 찾아, AI 서비스로 풀어냈습니다

내가 직접 한 일

- '잘 찍었나' 판정 대신 '고른 사진과의 차이 줄이기'로 문제 재정의
- 유사 서비스 비교로 비어 있던 단계(찍는 순간의 행동 안내) 정의
- 원인 순서 · 한 번에 하나 안내 정책과 자동 촬영 조건 설계
- 160문항 설문 설계로 규칙을 사용자 기준에서 검증
- 1초 로그 구조 설계와 관리자 화면 재생 분석 추가로 운영 개선 루프
- iOS 실시간 안내 구현 · TestFlight 배포 · 사업계획서 작성

결과

대상

중앙대 실전창업교육 F.A.S.T (2026.01)

우수상

중앙대 AI · SW 융합 우수성과 발표대회
(2026.09, 패션 리뷰 사진 연구)

우수 프로젝트

한이음 드림업 창의도전형 (2026)

489건 · 약 200명

실기기 세션 기록 · 전시 체험

다음 단계

레퍼런스를 고르지 않아도
AI가 구도 3안을 계산해
주는 바로찍기를 서비스로
내기