

발표_GhiVideo_좌표투영과_측점기반재생

GhiVideo 핵심 기술 두 가지 (발표 자료)

드론 주행영상 플레이어 **GhiVideo** 의 두 가지 핵심 기술을 초등학생도 이해할 수 있게 그림으로 설명합니다. ① 드론 좌표를 영상에 딱 맞추기(좌표 투영) · ② 시간이 아니라 '측점'으로 움직이는 재생 바

0. 한눈에 — GhiVideo는 뭐가 특별한가요?

보통 영상 플레이어는 "그냥 영상 재생"이 끝이에요. GhiVideo는 여기에 두 가지 마법을 더했어요.

① 좌표 투영

지도 속 터널·다리 위치를
영상 화면의 정확한 자리에
이름표로 딱 붙이기



② 측점 기반 재생 바

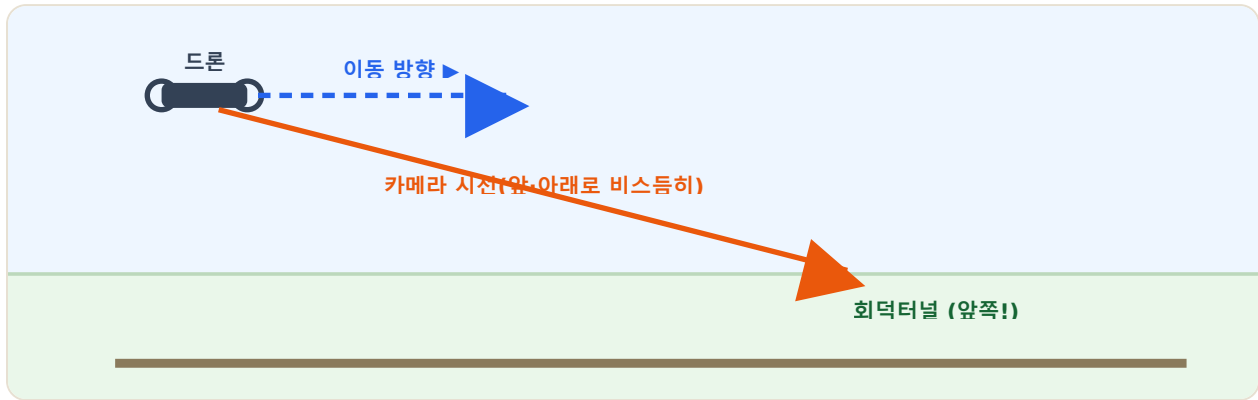
시간(0:00) 대신
철도 '측점(160k130)'으로
찾아가는 재생 바



Part 1. 드론 좌표를 영상에 '딱' 맞추기

1-1. 풀고 싶은 문제

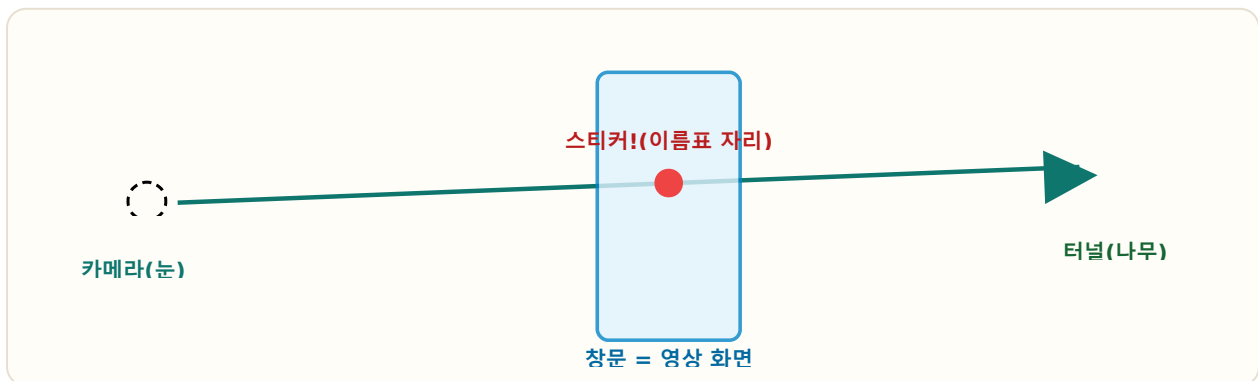
드론이 **앞으로 날며** 앞쪽 땅을 비스듬히 찍어요. 이 영상 위에 "**여기가 회덕터널!**" 이름표를 붙이고 싶어요. 드론이 계속 움직이니깐 이름표도 **줄줄 따라다녀야** 해요.



드론은 앞으로 날며 앞쪽 땅을 비스듬히 본다. 터널은 드론 바로 밑이 아니라 '앞쪽'에 찍힌다.

1-2. 핵심 아이디어 — '창문에 스티커 붙이기'

카메라(눈)에서 터널로 선을 긋고, 그 선이 화면(창문)을 뚫는 점에 이름표를 붙여요.



1-3. 실제로는 3단계로 계산해요



결과: "터널은 화면 가로 60%, 세로 70% 지점" → 거기에 이름표를 딱!

단계	진짜 기술	쉽게 말하면
① 미터로 바꾸기	proj4 · EPSG:5186(한국 TM)	동근 지구를 평평한 미터 지도로 펴기
② 방향 돌리기	회전행렬(yaw·pitch·roll)	카메라 기울기를 숫자표로 한 번에 적용

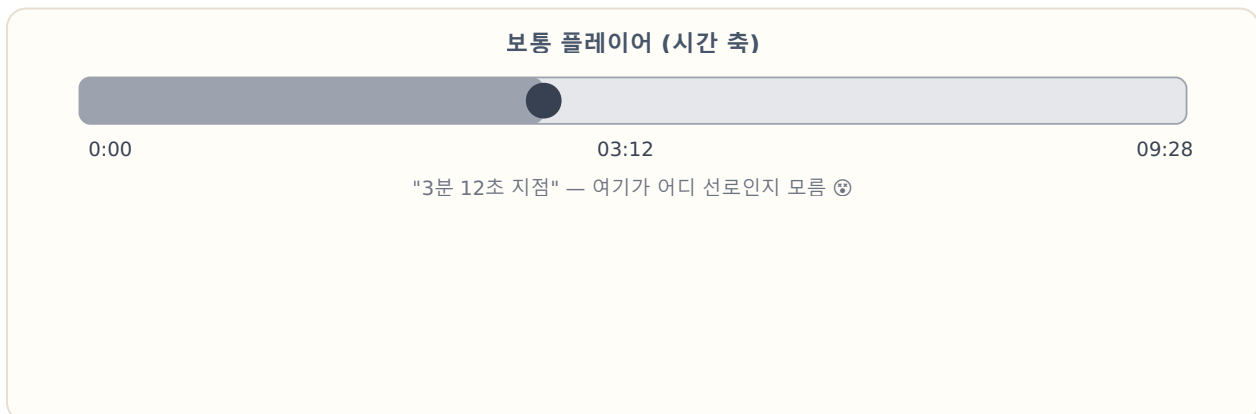
단계	진짜 기술	쉽게 말하면
③ 화면 점 찾기	핀홀 카메라 모델·초점거리	바늘구멍 사진기 원리로 점 찍기
(+) 부드럽게	보간 · EMA 평활	30장 사이 채우고 드론 떨림 없애기

원래 파이썬 실험 프로그램(`advanced_tuner_v2.py`)의 계산을 웹 브라우저(TypeScript)로 이식 → 설치 없이 실행.

Part 2. 시간이 아니라 '측점'으로 움직이는 재생 바

2-1. 무엇이 다른가요?

보통 플레이어의 아래 막대는 **시간(0:00 ~ 끝)** 기준이에요. 그런데 철도 점검은 "**몇 분**"이 아니라 "**몇 측점**"으로 말해요. 그래서 GhiVideo는 막대를 **측점(측점값, 예: 160k130)** 기준으로 바꿨어요.



GhiVideo (측점 축) 160k 회덕터널 회덕천교 대전조차장신탄진

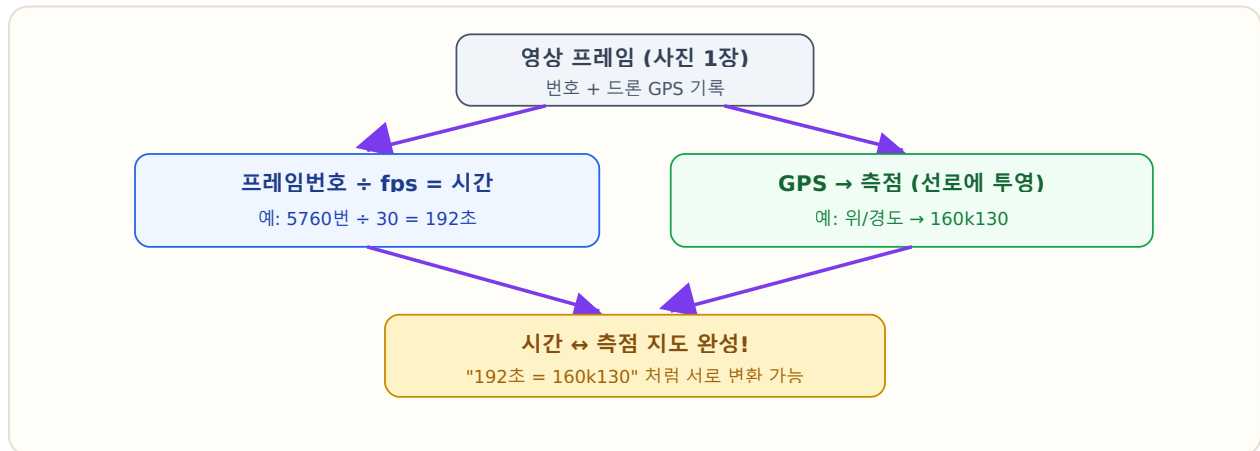
시간 축은 "몇 분"만 안다. 측점 축은 "지금 어느 선로 위치(측점)"와 "무슨 구조물 근처"인지 바로 보인다.

2-2. 측점이 뭐예요?

철도에서 **출발점부터 몇 미터 왔는지**를 나타내는 '거리 이정표'예요. `160k130` = 출발점에서 **160km + 130m** 지점. (도로의 'km 표지판'과 똑같아요!)

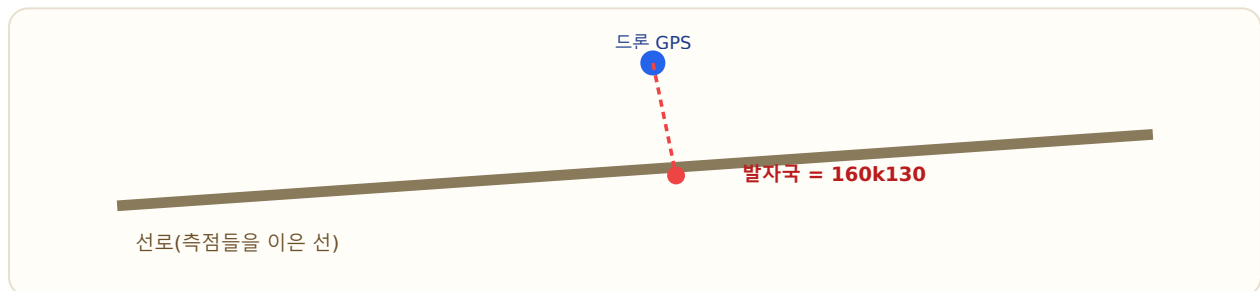
2-3. 어떻게 만들었나 — "시간↔측점 지도"

비밀은 드론 GPS예요. 영상의 매 프레임(사진 한 장) 마다 두 가지를 계산해 짝지어 줘요:



프레임마다 (시간)과 (측점)을 계산해 짝지으면, 시간↔측점을 자유롭게 오갈 수 있다.

GPS를 측점으로 바꾸는 법(②): 드론 GPS 점을 선로(측점들을 이은 선)에 수직으로 투영 떨어뜨려, 그 발자국이 몇 측점인지 읽어요. (코드: `projectToChain` / `projectChainage`)

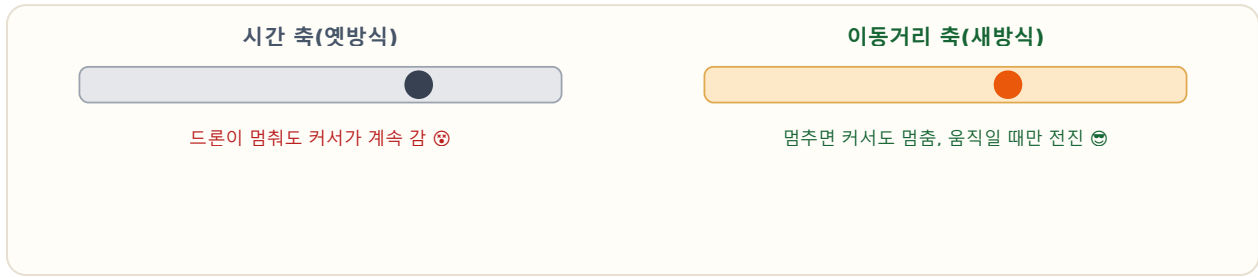


드론이 선로 바로 위가 아니어도, 수직으로 내린 발자국 위치로 '측점값'을 정확히 읽는다.

2-4. 똑똑한 점 3가지

① 막대의 가로축 = '시간'이 아니라 '실제 이동 거리'

드론이 멈춰서 맴돌면 시간은 흐르지만 위치는 그대로예요. 이때 막대 커서가 계속 가면 이상하죠. 그래서 가로축을 실제 이동한 거리(측점 변화량 누적)로 만들었어요. → 멈추면 커서도 멈추고, 움직이면 전진.



② 방향을 색으로 — 측점이 늘면 주황, 줄면 하늘색

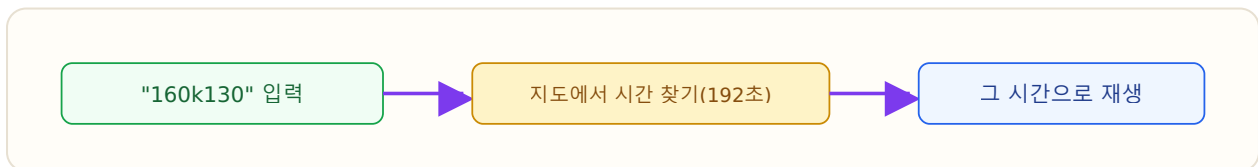
드론이 앞으로 가면(측점 증가) **주황**, 되돌아오면(측점 감소) **하늘색** 으로 칠해 **진행 방향**을 한 눈에 보여줘요.

③ 구조물을 제자리에 — 교량·터널·역은 '측점값'에 배치

교량·터널·역사를 각자의 **측점값 위치**에 표시해서, "이 구조물이 어디쯤인지" 막대만 봐도 알 수 있어요.

2-5. 측점으로 '찾아가기'(Seek)

측점값을 입력하거나 막대를 누르면 → **그 측점이 나오는 시간을 지도에서 찾아** → **그 시간으로 이동**해요. (시간↔측점 지도가 있으니 가능한 일!)



2-6. 핵심 기술 / 파일

하는 일	기술	파일 · 함수
GPS → 측점값	선분에 수직 투영	chainage.ts <code>projectToChain</code>
프레임 → 시간	fps 자동 산출	<code>VideoPlayer.tsx</code> <code>effectiveFps</code>
시간↔측점 지도	프레임별 precompute	StationBar.tsx
이동거리 측	측점 변화량 누적(정규화)	<code>StationBar.tsx</code> <code>frac</code>
방향 색·구조물 배치	측점값 기준 렌더	<code>stationbar/components/*</code>

마무리 — 한 문장씩

- **Part 1 (좌표 투영):** "드론이 어디서·어느 쪽을 보는지 알기 때문에, 지도 속 터널이 영상 화면 어디에 보일지 계산해 이름표를 딱 붙인다."
- **Part 2 (측점 기반):** "드론 GPS로 매 프레임의 측점을 계산해 시간↔측점 지도를 만들어, 시간이 아니라 측점으로 영상을 탐색한다."

두 기술 모두 **드론의 GPS·자세 데이터**를 똑똑하게 활용한 결과예요. 그래서 GhiVideo는 단순 영상이 아니라 "**선로 위 어디를 보고 있는지 아는**" **주행영상 플레이어**가 됩니다.