Real-Time Multi-Camera Tracking

1) What We Track (start with the “why”)[Problem]

Behaviors: loitering, tailgating, perimeter breach, object drop/left-behind, crowding, wrong-way movement.
Objects: people, weapons, backpacks/bags, vehicles/plates (optional), uniforms/hi-vis gear.
Attributes: clothing colors ("blue jacket"), accessories (hat/backpack), movement direction.

Pilot pick: choose top 3 behaviors + 2 objects per site to start. Expand once KPIs clear

2) Current Code → What It Already Does Well

Your pipeline (condensed):

- **Detection:** Ultralytics YOLO models (`people_model`, `gun_model`), fused for speed.
- **Tracking:** `supervision.ByteTrack` per stream; side-by-side output for two feeds.
- **Re-ID (prototype):** ResNet50 embedding head (`Identity()` final layer) + cosine similarity to link cam2 IDs to cam1.
- **Weapons heuristic:** keyword filter and a dedicated YOLO weapon model.
- **Moondream VLM:** per-crop Q&A to enrich labels/attributes (e.g., “is anyone holding a gun?”).

Strengths: tracks people on different camera streams, creates descriptions, can be used to ask questions
Gaps to close: identity stability across long time spans, attribute search from text, scale to thousands of cameras, fewer false positives.

Understanding decisions: Realtime analytics has trade offs with what we can achieve in short time and accuracy, there are other techniques that can be employed here to get more information like, passing images to VLM’s to get more detailed descriptions, or using bigger models for much better filtering but that would not work out in real time as for latency,

Now this extensible on most existing devices doesn't need any extra setup, but cost and compute gives more flexibility and adds features, like building a video rag, fine tune larger models with our data, openai gpt or google gemini for faster accurate responses

More improvements from production stand point

A. Faster + steadier identity

- **Detector:** use **YOLOv8n/8s** for person crops (already compatible).
- **Re-ID:** swap ResNet50 head for **OSNet_x0_25** (smaller, Re-ID-tuned).[For tracking across camera feeds]
- **Index:** store **L2-normalized 512D** embeddings in **FAISS** (Inner-Product search ≈ cosine).[matching people for nlp queries]
- **Centroid IDs:** maintain a running centroid per person-ID to stabilize assignment; keep **all per-frame vectors** in FAISS for recall.[quick retrieval]

B. Natural-language description search (Accurate and faster)

- **CLIP co-index:** during ingestion, compute **CLIP image features** for each crop → write to a second FAISS index.
- **Query time:** encode text (e.g., "man in blue jacket, red shoes"), shortlist with CLIP index → **re-rank** by Re-ID centroid to enforce identity.

C. Self-learning loop (Moondreamassisted or openai gpt)

- Use Moondream to generate **rich captions/attributes** for crops (color, items held, actions).
- Convert these to **training labels** to fine-tune YOLO (reduce false positives) and improve Re-ID on your camera domains.
- Route only **unusual/low-confidence events** to a larger VLM at the **central rig.**

D. Production tuning

- **Batching:** send multiple crops at once to the Re-ID/CLIP encoders.[speed]
- **Frame stride:** sample every **2–3 frames** to save compute without losing IDs.
- **Half precision (FP16):** for YOLO + Re-ID on supported GPUs.
- **Sharding:** per-site workers with **local FAISS shards**; optional IVF/PQ for very large corpora.

Use Cases

1) Tracking people in authorised locations and quickly identifying un authorised peron presence by matching faces on high quality cameras
2) Public place tracking for weapons, or other harmful object like near a school, we finetune our model from the data set we generate from the pipeline and then model can identify unusual behavior on a school playground,