



DOI: 10.24412/2181-144X-2025-1-129-148

Nishanov A.H., Narziyev N.B.,
Babadjanov E.S., Jabborzoda D. U.

BELGILANGAN HUDUDLARDA OBYEKTЛАRNI MAQSADLI KUZATISHNING MATEMATIK MODELИ VA ALGORITMIK TAHLILI.

Nishanov A.H. [0009-0006-4618-9437], **Babadjanov E.S.** [0009-0006-4618-9437]
Narziyev N.B. [0009-0006-4618-9437], **Jabborzoda D. U.** [0009-0006-4618-9437]

¹Muhammad al Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti professori,
E-mail nishanov_akhram@mail.ru

²Nukus davlat texnika universiteti, Kiberxavfsizlik injiniringi kafedrası mudiri,
E-mail elmurbes@gmail.com

³Muhammad al Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti ilmiy
izlanuvchisi, E-mail n.b.narziyev@gmail.com

⁴UNIVERSOFT IT masuliyati cheklangan jamiyati director maslahatchisi,
E-mail domkajb@gmail.com

Annotatsiya. Ushbu maqolada belgilangan hududlarda obyektlarni maqsadli kuzatish va boshqarishning matematik asoslari ishlab chiqilgan. Nufuzli jurnaldagi tegishli shlarni tizimli tahlil natijasida ularning aksariyati aniq predmet sohalaridagi vazifalarni hal qilishga qaratilgan bo'lib, lekin barcha asosiy vazifalarni yagona matematik model ichida birlashtirgan tadqiqot mavjud emasligi aniqlandi. Ushbu bo'shliqni to'ldirish maqsadida ishda kuzatish tizimining umumiy modeli taklif etiladi. Bunda kuzatish obyektlari, nazorat hududlari, kuzatish vositalari, vaqt va funksional operatorlarni yagona matematik framework ichida birlashtiradi. Xususan, 5 ta asosiy vazifalar (detection, embedding, matching, tracking va behavior analysis) formal matematik formulatsiya bilan tavsiflangan. Natijada vazifalarning konveyer (pipeline) sifatida integratsiyasi tahlil qilinib, xatolar tarqalishi muammosi formal ko'rsatilgan, ya'ni 5 ta 95% aniqlikdagi operator birgalikda 77.4% aniqlik beradi. Qayta aloqa (feedback) mexanizmlari va end-to-end optimallashtirish bu muammoni yumshatish yo'llari sifatida taklif etilgan.

Kalit so'zlar: intellektual kuzatish, matematik modellashtirish, Re-ID, MOT, metric learning, maqsad funksiyasi, embedding, Kalman filter.

Аннотация. В данной статье разработаны математические основы целевого наблюдения и управления объектами в заданных зонах. В результате системного анализа соответствующих работ в авторитетных журналах установлено, что большинство из них направлены на решение задач в конкретных предметных областях, однако отсутствуют исследования, объединяющие все основные задачи в рамках единой математической модели.

С целью устранения данного пробела в работе предложена обобщённая модель системы наблюдения. В ней объекты наблюдения, зоны контроля, средства наблюдения, время и функциональные операторы объединены в единую математическую структуру.

В частности, пять основных задач (обнаружение, embedding, сопоставление, отслеживание и анализ поведения) описаны с использованием формальных математических постановок. В результате проанализирована их интеграция в виде конвейера (pipeline), а также формально показана проблема накопления ошибок: например, пять операторов с точностью 95% в совокупности дают лишь 77,4% точности.

В качестве способов смягчения данной проблемы предложены механизмы обратной связи (feedback) и сквозная (end-to-end) оптимизация.

Ключевые слова: интеллектуальное наблюдение, математическое моделирование, Re-ID, MOT, metric learning, целевая функция, embedding, фильтр Калмана.

Abstract. This paper develops the mathematical foundations for targeted monitoring and control of objects within designated areas. A systematic analysis of relevant publications in high-impact journals shows

that most studies focus on solving problems within specific application domains, while there is a lack of research that integrates all core tasks into a unified mathematical model.

To address this gap, a generalized model of the monitoring system is proposed. The model unifies observation objects, control areas, sensing devices, time, and functional operators within a single mathematical framework.

In particular, five key tasks (detection, embedding, matching, tracking, and behavior analysis) are described using formal mathematical formulations. The integration of these tasks as a pipeline is analyzed, and the issue of error propagation is formally demonstrated: for instance, five operators each with 95% accuracy result in an overall accuracy of only 77.4%.

Feedback mechanisms and end-to-end optimization are proposed as approaches to mitigate this issue.

Keywords: intelligent surveillance, mathematical modeling, Re-ID, MOT, metric learning, objective function, embedding, Kalman filter.

Kirish

Zamonaviy urbanizatsiya jarayonlari jamoat joylarida xavfsizlikni ta'minlash masalasini keskin dolzarb qildi. Birlashgan Millatlar Tashkilotining so'ngi berilgan ma'lumotlariga ko'ra, dunyo aholisining 57% dan ortig'i shaharlarda istiqomat qiladi va bu ko'rsatkich 2050 yilga kelib 68% ga yetishi bashorat qilinmoqda [1]. Aeroportlar, temir yo'l vokzallari, savdo mamajmualari, stadionlar va boshqa jamoat joylari kuniga millionlab odamlarga xizmat ko'rsatadi.

Bugungi kunga kelib an'anaviy kuzatish usullari bu masshtabda samarali ishlamay qoldi. Tadqiqotlar shuni ko'rsatadiki, inson operatori bir vaqtda 4-6 ta monitor tasvirini samarali kuzata oladi, 20 daqiqadan so'ng e'tibor darajasi 30% ga pasayadi [2, 3]. Yuzlab kamerali tizimlarda inson resursi yetarli emas va ushbu jarayonlarni avtomatlashtirish zarur bo'ladi.

Intellectual kuzatish tizimining asosiy vazifalaridan birida berilgan H – nazorat hududi (geometriya, zonalar, o'tish qoidalari), V – kuzatish vositalari to'plami (kameralar, sensorlar) va $T = [t_0, t_1]$ – kuzatish vaqt oraliq parametrlari orqali aniqlangan obyektlardan O har birining id_i – identifikatori, τ_i – traektoriyasi, a_i – xatti-harakatlarini topish kerak. Bu quyidagicha matematik formallashtiriladi:

$$P: (H, V, T) \rightarrow \{(o_i, \tau_i, a_i)\}_{i=1}^N.$$

Bu umumiy vazifa aniqlash (detection), xususiyat ajratish (feature extraction), identifikatsiya (identification), kuzatib borish (tracking) va xatti-harakat tahlili (behavior analysis) kabi bir nechta kichik vazifalarga bo'linadi. Har bir vazifa o'z algoritmik yechimlariga ega bo'lib, o'z o'rnida ularni yagona tizim sifatida birlashtirish alohida muammo hisoblanadi.

1.2. Mavjud tadqiqotlar tahlili

Intellectual kuzatish sohasida so'nggi o'n yilda juda katta progressiv o'zgarishlar kuzatilmog'ida. Jumladan, chuqur o'qitish (deep learning) usullari deyarli barcha vazifalarda an'anaviy usullarni ortda qoldirdi. Mavjud tadqiqotlarni sistematik tahlil qilish uchun Web of Science, Scopus va IEEE Xplore bazalarida qidiruv o'tkazildi. Kalit so'zlar sifatida "intelligent surveillance", "multi-object tracking", "person re-identification", "behavior analysis" va ularning kombinatsiyalari ishlatildi. 2014-2024 yillar oralig'ida jami 1,480 dan ortiq maqola topildi. Saralash mezonlari quyidagicha belgilandi: (1) faqat peer-reviewed jurnal va konferensiya maqolalari, (2) chuqur o'rganishga asoslangan usullar, (3) real vaqt tizimlariga qo'llanilishi mumkin. Natijada 156 ta maqola batafsil ko'rib chiqildi.

Tahlil natijasi shuni ko'rsatdiki, mavjud tadqiqotlarning aksariyati alohida vazifalarni hal qilishga qaratilgan. Barcha beshta vazifani (aniqlash, embedding, moslashtirish, kuzatish, xatti-harakat tahlili) yagona matematik model ichida formal tavsiflagan ish topilmadi.

Mavjud tadqiqotlarning vazifalar bo'yicha taqsimoti

Vazifa	Maqolalar soni	Ulushi	Asosiy usullar
Detection	412	28%	YOLO, Faster R-CNN, DETR
Re-ID / Embedding	298	20%	ArcFace, Triplet, TransReID
Tracking (MOT)	347	23%	SORT, DeepSORT, ByteTrack
Behavior Analysis	231	16%	ST-GCN, I3D, Autoencoder
Ikki va undan ortiq vazifa birgalikda	192	13%	JDE, FairMOT
Barcha 5 vazifa yagona modelda	0	0%	–

Jadvaldan ko'rinadiki, tadqiqotlarning 87% alohida vazifaga qaratilgan. 13% ikki vazifani birlashtirgan (masalan, JDE [22] da detection va embedding). Biroq barcha beshta vazifani yagona matematik framework da birlashtirgan tadqiqot topilmadi. Ushbu maqola aynan shu bo'shliqni to'ldirishga qaratilgan.

Obyektni aniqlash. 2014 yilda Girshick va hammualiflari R-CNN arxitekturasini taklif qildilar – birinchi marta CNN obyekt aniqlashda muvaffaqiyatli qo'llanildi [4]. Ren va boshqalar Faster R-CNN da Region Proposal Network (RPN) ni qo'shib, tezlik va aniqlikni sezilarli oshirdilar [5]. Parallel ravishda Redmon va hammualiflari butunlay boshqa yondashuv – YOLO (You Only Look Once) ni taklif qildilar, bu usul tasvirni gridga bo'lib bir o'tishda aniqlaydi [6]. YOLOV8 MS COCO datasetida 53.9% mAP ko'rsatib, real vaqt aniqlashda eng yaxshi natijalarga erishdi [7].

Shaxsni identifikatsiya. Yuzni tanib olish sohasida Taigman va hammualiflari DeepFace da 97.35% aniqlikka erishdilar – bu inson darajasiga yaqin natija [8]. Deng va boshqalar ArcFace loss funksiyasini taklif qildilar – angular margin qo'shish sinflar orasidagi chegarani kengaytirdi va LFW datasetida 99.83% aniqlik ko'rsatildi [9]. Person Re-ID sohasida He va hammualiflari TransReID arxitekturasida Vision Transformer ni qo'llab, Market-1501 da 95.2% Rank-1 aniqlik ko'rsatdilar [10]. Ye va boshqalar keng qamrovli survey da ReID usullarini tizimlashtirib, ochiq muammolarni aniqladilar [11].

Ko'p obyektни kuzatish. Bewley va hammualiflari SORT (Simple Online and Realtime Tracking) algoritmidagi Kalman filter va Hungarian algoritmini birlashtirib, sodda ammo samarali yechim yaratdilar [12]. Wojke va boshqalar DeepSORT da appearance embedding qo'shib, ID switching muammosini 45% ga kamaytirishdi [13]. Zhang va hammualiflari ByteTrack da past ishonchli aniqlashlarni ham ishlatib, MOT17 da 80.3 MOTA ko'rsatdilar [14].

Xatti-harakat tahlili. Yan va hammualiflari ST-GCN (Spatial-Temporal Graph Convolutional Network) da inson skeletini graf sifatida modellashtirdilar [15]. Bo'g'imlar tugunlar, suyaklar qirralar sifatida ifodalanib, spatial va temporal convolution qo'llanildi. NTU RGB+D datasetida 81.5% aniqlik ko'rsatildi.

Yuqoridagi tadqiqotlar alohida vazifalarni hal qilishga qaratilgan. Ularni yagona matematik framework ichida birlashtirish, vazifalar orasidagi o'zaro ta'sirni formal tavsiflash va tizimni umumiy optimallashtirish kam o'rganilgan.

Tadqiqot maqsadi va vazifalari

Ushbu maqolaning maqsadi belgilangan hududlarda obyektlarni maqsadli kuzatish uchun yagona matematik model ishlab chiqish va asosiy algoritmik muammolarni ushbu model doirasida tahlil qilishdir. Mazkur maqsadga erishish uchun quyidagi vazifalar belgilab olindi:

1. Kuzatish tizimining umumiy matematik modelini ishlab chiqish



2. Kuzatish obyektlarining formal tavsifini berish
3. Asosiy vazifalar uchun matematik formulatsiyalarni keltirish
4. Vazifalarning yagona pipeline sifatida integratsiyasini ko'rsatish

Intellectual kuzatish sohasida so'nggi o'n yilda juda katta progressiv o'zgarishlar kuzatilmoqda. Jumladan, chuqur o'qitish (deep learning) usullari deyarli barcha vazifalarda an'anaviy usullarni ortda qoldirdi.

Intellectual kuzatish tizimini formal tavsiflash uchun uning barcha komponentlarini yagona matematik struktura ichida birlashtirish zarur. Bunday tizim qanday obyektlarni kuzatishi, qaysi hududda ishlashi, qanday vositalardan foydalanishi, qancha vaqt davomida kuzatishi va qanday funksiyalarni bajarishi kerakligini aniq belgilash lozim. Ushbu komponentlarni hisobga olib, intellektual kuzatish tizimi quyidagi beshlik sifatida aniqlanadi:

$$S = (O, H, V, T, \Phi)$$

bu yerda O – kuzatish obyektlari to'plami, H – nazorat hududlari to'plami, V – kuzatish vositalari to'plami, T – kuzatish vaqt oralig'i, $\Phi = \{D, E, M, T, A\}$ – funksional operatorlar to'plami

Bu model universallik xususiyatiga ega bo'lib, turli hududlar (aeroport, savdo markazi, qamoqxona) uchun komponentlarni moslashtirish orqali qo'llaniladi.

Kuzatish obyekti deganda tizim tomonidan kuzatiladigan va identifikatsiya qilinadigan birlik tushuniladi. Har bir obyekt noyob identifikatorga ega bo'lib, uning fazoviy joylashuvi, tashqi ko'rinishi va holati vaqt o'tishi bilan o'zgaradi. Shu sababli obyekt o_i vaqtga bog'liq to'rtlik sifatida aniqlanadi:

$$o_i(t) = (id_i, x_i(t), a_i(t), s_i(t))$$

bu yerda id_i – obyektning noyob identifikatori, $x_i(t) \in R^3$ – t vaqtidagi fazoviy koordinatalar (x, y, z), $a_i(t) \in R^d$ – atributlar vektori (d – atributlar soni), $s_i(t)$ – obyekt holati.

Mazkur to'rtlik parametrlar ham ichki cheklangan to'plam qiymatlarini qabul qiladi. Jumladan, atributlar vektori obyektning vizual va semantik xususiyatlarini ifodalaydi. Xususan, shaxs obyekti uchun $a_{person} = (a_{bio}, a_{app}, a_{beh})$ namuna keltirilib, bu a_{bio} – biometrik (sekin o'zgaruvchi), a_{app} – tashqi ko'rinish (tez o'zgaruvchi) va a_{beh} – xatti-harakat (dinamik) atributlaridan iborat.

Obyekt holati ham cheklangan to'plam qiymatino oladi: $s_i(t) \in S = \{active, occluded, lost, exited\}$. Holatlar orasidagi o'tishlar quyidagi qoidalarga bo'ysunadi: obyekt qisman yoki to'liq yopilganda active \rightarrow occluded; obyekt qayta ko'ringanda occluded \rightarrow active; yopilish muddati θ_{lost} dan oshganda occluded \rightarrow lost; obyekt hududdan chiqqanda active \rightarrow exited; yo'qolgan obyekt qayta aniqlanganda (ReID orqali) lost \rightarrow active.

Traektoriya o_i obyektning $x_i(t)$ koordinatalari vaqt bo'yicha to'plami (fazoviy holatlari ketma-ketligi) $\tau_i = \{(x_i(t), t) : t \in T_i\}$ shaklida ifodalanib, bu yerda $T_i \subseteq T$ – obyekt kuzatilgan vaqt intervallari to'plami. Traektoriya uzluksiz (kamera ichida) yoki uzulishli (kameralararo o'tishda) bo'lishi mumkin.

Nazorat hududi kuzatish tizimi faoliyat ko'rsatadigan cheklangan geografik maydon bo'lib, har bir hudud o'z geometrik chegaralariga, ichki zonalarga (masalan, kirish, chiqish, cheklangan) va zonalar orasidagi o'tish qoidalarga hamda boshqa muhim hudud parametrlariga ega. Ushbu xususiyatlarni hisobga olib, nazorat hududi h_j quyidagi tuzilma bilan aniqlanadi:

$$h_j = (G_j, Z_j, R_j, p_j)$$

bu yerda G_j – hudud geometriyasi (3D polygon yoki mesh), $Z_j = \{z_1, z_2, \dots, z_K\}$ – ichki zonalar to'plami, $R_j \in \{0, 1\}^{K \times K}$ – zonalar orasidagi o'tish matritsasi, p_j – hudud parametrlari (sig'im, xavfsizlik darajasi).

Hudud bir nechta zonalarga bo'linadi, har bir zona o'z funksional vazifasiga ega. Masalan, aeroportda umumiy kutish zonasi, xavfsizlik tekshiruvchi zonasi, boarding zonasi

mavjud. Hududlar $z_l = (G_l, level_l, capacity_l, allowed_l)$ tuzilmada belgilanib, bunda har bir zona o'z xavfsizlik darajasi ($level_l$), maksimal sig'imi ($capacity_l$) va kirish huquqidagi ($allowed_l \subseteq O$) obyektlar ro'yxatiga ega.

Nazorat hududida obyektlar bir zonadan boshqasiga o'tish cheklovlariga (erkin harakat, o'tishga ruxsat va taqiqlangan) ega bo'ladi. Masalan, aeroportda yo'lovchi umumiy zonadan to'g'ridan-to'g'ri uchish zonasiga o'ta olmaydi. Bu cheklovlarni formal ifodalash uchun o'tish matritsasi R kiritiladi, bu qaysi zonalardan qaysilarga o'tish mumkinligini belgilaydi:

$$R[z_a, z_b] = \begin{cases} 1, & \text{agar } z_a \rightarrow z_b \text{ ruxsat etilgan} \\ 0, & \text{aks holda.} \end{cases}$$

O'tish matritsasi anomaliya aniqlashda muhim. Agar o_i obyekt z_a zonasidan z_b zonasiga o'tgan bo'lsa, bu o'tish matritsasi R orqali tekshiriladi:

$$Anomaly_{zone}(o_i, t) = 1[R[zone(o_i, t - \Delta t), zone(o_i, t)] = 0].$$

Kuzatish vositasi tizimning "ko'zi" bo'lib, u muhitdan vizual ma'lumot oladi. Kamera tasvirining sifati va qamrovi uning joylashuvi, optik parametrlari va texnik xususiyatlariga bog'liq. Shu sababli kuzatish vositasi v_k quyidagi parametrlar to'plami sifatida aniqlanadi:

$$v_k = (p_k, K_k, R_k, FOV_k, f_k, \rho_k)$$

bu yerda $p_k \in R^3$ – kameraning fazoviy joylashuvi, $K_k \in R^{3 \times 3}$ – ichki parametrlar matritsasi (intrinsic), $R_k \in SO(3)$ – orientatsiya matritsasi (rotation), $FOV_k = (\theta_h, \theta_v)$ – gorizont va vertikal ko'rish burchagi, f_k – kadr tezligi (FPS), ρ_k – piksel zichligi (resolution).

Ichki parametrlar matritsasi (intrinsic) quyidagicha ifodalanadi:

$$K_k = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix},$$

bu yerda f_x, f_y – fokal uzunlik (pikselda), c_x, c_y – asosiy nuqta koordinatalari.

Ko'rish maydoni (Field of View) kameraning qamrab oladigan hududini aniqlaydi. Bu frustum (uchi parallel tekislik bilan kesigan konus/piramida) geometriyasi bilan ifodalanadi:

$$FOV_k = (x, y, z): |\text{atan2}(x, z)| < \frac{\alpha_h}{2}, |\text{atan2}(y, z)| < \frac{\alpha_v}{2}, z > 0$$

Kameralar tarmoqi yo'naltirilgan graf sifatida ifodalanadi:

$$G_V = (V, E_V)$$

bu yerda V – kameralar to'plami, $E_V = \{(v_a, v_b, \Delta t_{ab})\} \subseteq V \times V \times R^+$ – qirralar to'plami. Har bir qirra kameralar orasidagi o'tish imkoniyati va kutilgan vaqtni bildiradi. Bu graf kameralararo tracking uchun muhim – obyekt v_a dan chiqib, v_b da paydo bo'lishi kutilgan vaqtni Δt_{ab} belgilaydi.

Intellektual kuzatish tizimi bir nechta ketma-ket bajariladigan funksiyalardan tashkil topadi. Avval tasvirda obyektlar aniqlanadi, so'ng ularning xususiyatlari ajratiladi, bazadagi ma'lumotlar bilan solishtiriladi, vaqt bo'yicha kuzatib boriladi va nihoyat xatti-harakati tahlil qilinadi. Tizimning asosiy funksiyalari operatorlar to'plami $\Phi = \{D, E, M, T, A\}$ sifatida quyidagicha aniqlanadi:

- Aniqlash operatori (Detection): $D: I \rightarrow (b_j, c_j, p_j)_{j=1}^{N_d}$, bu yerda $I \in R^{H \times W \times 3}$ – kirish tasviri, $b_j = (x, y, w, h)$ – chegaralovchi to'rtburchak, $c_j \in C$ – obyekt klassi, $p_j \in [0, 1]$ – ishonch darajasi (confidence).

- Embedding operatori (Feature Extraction): $E: I_{crop} \rightarrow e \in R^d, |e|_2 = 1$, bu yerda I_{crop} – aniqlangan obyektning qirqilgan tasviri, e – normallashtirilgan embedding vektor, d – embedding o'lchami (odatda 128, 256, 512 yoki 2048).

- Moslik operatori (Matching): $M: (e_{probe}, G) \rightarrow (id^*, s^*)$, bu yerda e_{probe} – so'rov embeddingi, $G = (id_i, e_i)_{i=1}^N$ – galereya (bazadagi embeddinglar), id^* – topilgan identifikator, s^* – o'xshashlik balli.

•Kuzatish operatori (Tracking): $T: (B_t, E_t, T_{t-1}) \rightarrow T_t$, bu yerda B_t – t vaqtdagi aniqlashlar, E_t – ularning embeddinglari, T_{t-1} – oldingi tracklar, T_t – yangilangan tracklar to'plami.

•Tahlil operatori (Behavior Analysis): $A: \tau_i \rightarrow (action_i, prob_i)$, bu yerda τ_i – obyekt traektoriyasi, $action_i \in A$ – aniqlangan harakat/xatti-harakat klassi, $prob_i$ – ishonch darajasi.

Intellectual kuzatish tizimining umumiy maqsadi barcha obyektlarni maksimal aniqlik bilan aniqlash, identifikatsiya qilish va kuzatishdir. Bu maqsad quyidagi funksional sifatida ifodalanadi:

$$J(S) = \sum_{i=1}^{|O|} [1_{detected}(o_i) \cdot 1_{identified}(o_i) \cdot Q(\tau_i)],$$

bu yerda $1_{detected}(o_i)$ – obyekt aniqlanganmi, $1_{identified}(o_i)$ – to'g'ri identifikatsiya qilinganmi, $Q(\tau_i)$ – traektoriya sifati. Traektoriya sifatini baholashda tizim tomonidan qurilgan traektoriya (τ_i) haqiqiy bilan (etalon) traektoriya (τ_i^{gt}) quyidagicha solishtiriladi:

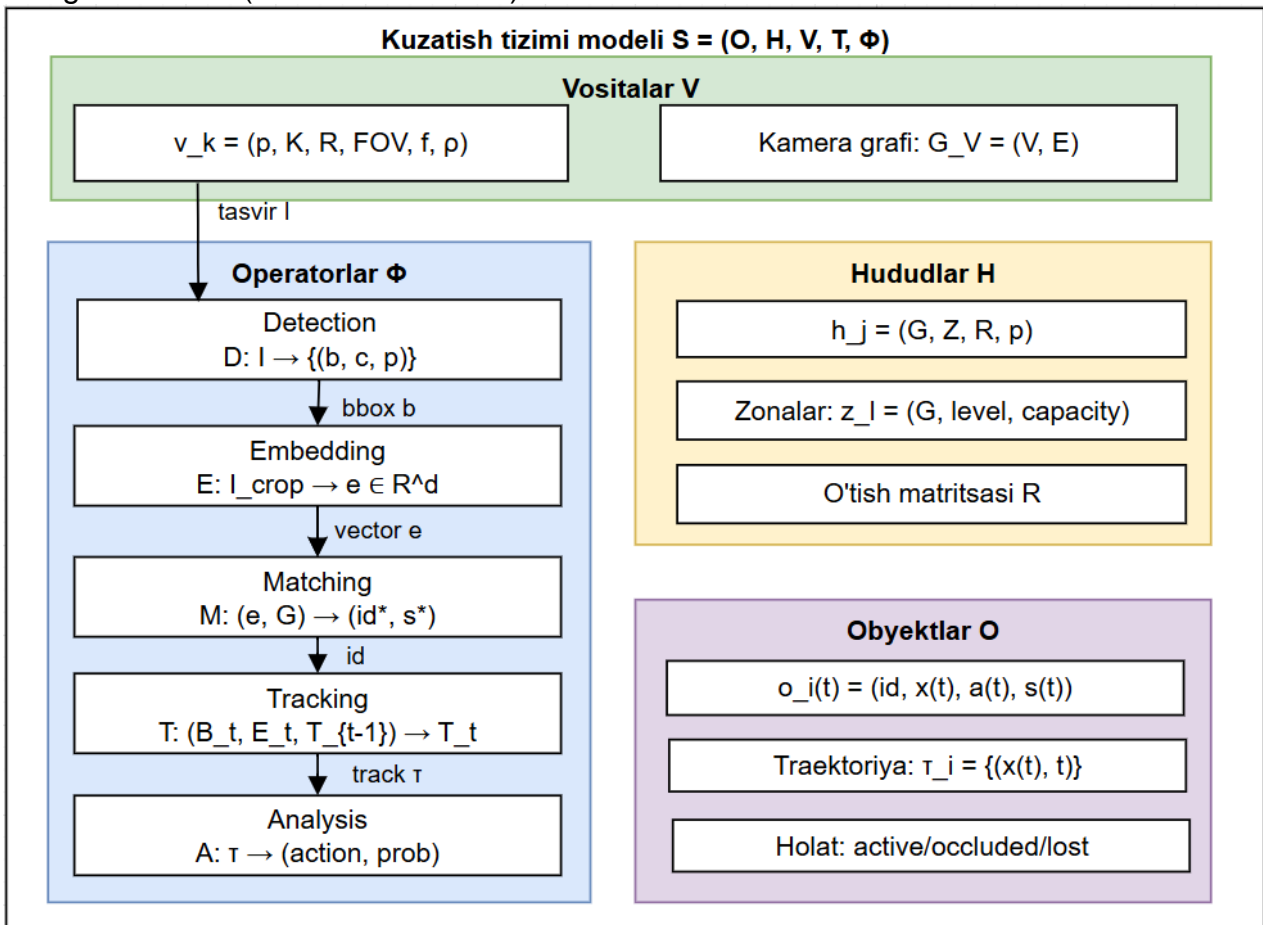
$$Q(\tau_i) = \frac{|\tau_i \cap \tau_i^{gt}|}{|\tau_i \cup \tau_i^{gt}|} \cdot \left(1 - \frac{N_{switch}}{|\tau_i|}\right),$$

bu yerda N_{switch} – ID almashinuvlar soni. Birinchi ko'paytuvchi had fazoviy to'g'rilikni (IoU), ikkinchi had esa identifikator barqarorligini o'lchaydi.

Tizimni optimallashtirishda operatorlar parametrlarini shunday tanlash kerakki, unda $J(S)$ maqsad funksiyasi maksimal bo'lsin:

$$\Phi^* = \arg \max_{\Phi} J(S)$$

Amalda bu optimallashtirish har bir operator uchun alohida loss funksiyalari orqali amalga oshiriladi (4-bo'limda batafsil).



1-rasm. Kuzatish tizimining umumiy arxitekturasi

Ushbu bo'limda kuzatish tizimining umumiy matematik modeli $S=(O, H, V, T, \Phi)$ taklif etildi va uning komponentlari formal tavsiflandi. Biroq model doirasidagi obyektlar (shaxs, transport, narsa) o'ziga xos xususiyatlarga ega bo'lib, ularning har biri alohida modellashtirish talab qiladi. Keyingi bo'limda kuzatish obyektlarining batafsil formal tavsifi beriladi.

Har bir obyekt turi o'ziga xos xususiyatlarga ega va bu xususiyatlar kuzatish algoritmlariga bevosita ta'sir qiladi. Endi bo'limda asosiy obyekt turlarining tabiati, ularni kuzatishdagi qiyinchiliklar va mavjud yechimlar tahlil qilinadi.

Intellectual kuzatish tizimlari odamlar, transport vositalari, narsalar kab turli xil obyektlar bilan ishlaydi. Har bir obyekt turi o'ziga xos atributlar strukturasi va kuzatish algoritmlariga ega. Shuning uchun ularni alohida ko'rib chiqish maqsadga muvofiq. Shu sababli kuzatish obyektlari to'plami O quyidagi kichik to'plamlarga bo'linadi: $O = O_{person} \cup O_{vehicle} \cup O_{object}$. Har bir kichik to'plam o'ziga xos atributlar strukturasi va kuzatish algoritmlariga ega.

Shaxs intellektual kuzatish tizimlarining asosiy obyekti. Boshqa obyektlardan farqli ravishda shaxslar murakkab va ko'p qirrali kuzatish talablarini qo'yadi. Shaxs obyekti kengaytirilgan model bilan tavsiflanadi:

$$o_{person} = (id, x(t), a_{bio}, a_{app}, a_{beh}, s(t)).$$

Biometrik atributlar shaxsning o'zgarimas yoki sekin o'zgaruvchi xususiyatlarini ifodalaydi:

$$a_{bio} = (e_{face}, e_{body}, age, gender, height),$$

bu yerda $a_{face} \in R^{d_f}$ – yuz embeddingi ($d_f=128$ yoki 512), $e_{body} \in R^{d_b}$ – tana embeddingi ($d_b=256$ yoki 2048), $age \in [0, 100]$ – yosh, $gender \in \{0, 1\}$ – jins, $height \in [0.5, 2.5]$ – bo'y (metrda). Yuz embeddingi a_{face} – shaxsni identifikatsiya qilishning eng ishonchli xususiyati. Deng [9] ArcFace da ResNet-100 backbone bilan $d_f = 512$ o'lchamli embedding hosil qilgan.

Tashqi ko'rinish atributlari vaqt bilan o'zgaruvchi xususiyatlarni ifodalaydi:

$$a_{app}(t) = (c_{upper}, c_{lower}, c_{hair}, accessories),$$

bu yerda $c_{upper}, c_{lower}, c_{hair} \in R^3$ – kiyim va soch ranglari (RGB), $accessories \subset \{\text{ko'zoynak, shlyapa, sumka, niqob, ...}\}$ – aksessuarlar to'plami. Shaxs ReID algoritmlari asosan a_{app} ga tayanadi. Ammo kiyim o'zgarsa – aniqlik keskin pasayadi. Ye va boshqalar [11] ko'rsatdiki, kiyim almashtirish 15-25% aniqlik yo'qotishiga olib keladi.

Xatti-harakat atributlari dinamik xususiyatlarni ifodalaydi:

$$a_{beh}(t) = (v(t), \theta(t), action(t), pose(t))$$

bu yerda $v(t) = ||x'(t)||$ – tezlik (m/s), $\theta(t) = atan2(y', x')$ – harakat yo'nalishi, $action(t) \in A$ – joriy harakat klassi, $pose(t) \in R^{J \times 3}$ – skelet pozasi (J bo'g'im)

Transport vositalari shaxslardan tubdan farq qiladi – ular nisbatan barqaror ko'rinishga ega va identifikatsiya qilish osonroq ko'rinadi. Biroq o'ziga xos muammolari mavjud. Transport vositasi quyidagi model bilan tavsiflanadi:

$$o_{vehicle} = (id, x(t), a_{static}, a_{dynamic}, s(t))$$

Statik atributlar: $a_{static} = (turi, davlat raqami, vizual xususiyatlar)$, bu yerda $turi \in \{\text{engil, yuk, avtobus, mototsikl, ...}\}$, vizual xususiyatlar (rang, model, shakl) raqam ko'rinmaganida ishlatiladi. Davlat raqami eng ishonchli identifikator. ALPR (Automatic License Plate Recognition) tizimlar 99%+ aniqlik ko'rsatadi. Liu va hammualiflari [17] VeRi-776 datasetida vehicle ReID uchun 79.3% mAP ko'rsatdilar.

Dinamik atributlar: $a_{dynamic}(t) = (v(t), a(t), \theta(t), lane(t))$, bu yerda v – tezlik, a – tezlanish, θ – yo'nalish, $lane$ – harakat chizig'i.

Narsalar (sumkalar, jomadonlar, paketlar) o'z-o'zidan qiziq emas. Ular shaxslarga bog'liq holda ahamiyat kasb etadi. Asosan bu narsa kimniki va u qolib ketganmi kabi xavfsizlik nuqtai nazaridan muhim:

$$o_{object} = (id, x, a_{obj}, owner, t_{last}),$$

bu yerda $owner \in O_{person} \cup \{null\}$ – egasi, t_{last} – oxirgi harakat vaqti.

Aksariyat xatali vaziyatlarda qolib ketgan sumka yoki jomadon orqali amalga oshiriladi. Bunda tizim agar narsa ma'lum vaqt davomida harakatsiz tursa va yonida egasi ko'rinmasa, ogohlantirish berish mantiqida ishlaydi. Bu vazifaning murakkabligi shundaki, narsa va egasini bog'lash kerak. Misol uchun shaxs sumkani qo'ydi va 2 metr uzoqlashsa, bu normal. Amm 10 metr uzoqlashsa va qaytmasa, bu shubhali. Qolib ketgan narsani (abandoned) aniqlash qoidasi quyidagicha:

$$abandoned(o) = 1[owner = null] \cdot 1[t - t_{last} > \theta_{aband}],$$

bu yerda θ_{aband} – chegara vaqti (odatda 30-60 sek.).

Kuzatish tizimi obyektlarni alohida emas, o'zaro bog'liqlikda ko'rishi kerak. Misol uchun ikki shaxs birga yurisa, ular tanish bo'lishi, katta guruh to'planshi esa miting yoki muammo bo'lishi mumkin. Bu munosabatlarni tushunish tizimni "aqli" qiladi. Obyektlar orasida egalilik va guruh kabi turli munosabatlar mavjud.

Egalilik munosabati $owns: O_{person} \times O_{object} \rightarrow \{0,1\}$ bilan belgilanadi. Agar p-shaxs va o-narsa uzoq vaqt davomida yaqin masofada birga harakatlansa, narsa shu shaxsga tegishli deb hisoblanadi. Bu fazoviy yaqinlik va vaqtli korrelyatsiya orqali aniqlanadi:

$$owns(p, o) = \mathbb{1} \left[\frac{1}{|T|} \sum_{t \in T} |x_p(t) - x_o(t)| < \epsilon_{own} \right]$$

bu yerda $|x_p - x_o|$ – t-vaqtda shaxs va narsa orasidagi Evklid masofasi, $|T|$ – kuzatish davomiyligi, ϵ_{own} – masofa chegarasi (odatda 1-1.5 metr). Agar o'rtacha masofa chegaradan kichik bo'lsa, $\mathbb{1}[\cdot] = 1$ (egalilik bor), aks holda 0.

Guruh munosabati oddiy ko'rinsa ham matematik jihatdan murakkab. Qachon odamlar guruhi "guruh" hisoblanadi? Ge va hammualiflari [23] ikki shartni taklif qildilar: fazoviy yaqinlik (bir-biridan 2 metr ichida) va harakat sinxronligi (bir yo'nalishda, bir tezlikda). Bu shartlar sodda, ammo samarali – 82% guruhlar to'g'ri aniqlanadi. Guruh munosabati $group: 2^{O_{person}} \rightarrow 0,1$ orqali berilib, shaxslar guruhi fazoviy yaqinlik va harakat sinxronligi orqali aniqlanadi:

$$group(P) = \mathbb{1} \left[\max_{p_i, p_j \in P} |x_{p_i} - x_{p_j}| < \epsilon_g \right] \cdot \mathbb{1} \left[corr(v_{p_i}, v_{p_j}) > \rho_g \right],$$

$$corr(v_{p_i}, v_{p_j}) = \frac{\sum_t (v_{p_i}(t) - \bar{v}_{p_i})(v_{p_j}(t) - \bar{v}_{p_j})}{\sqrt{\sum_t (v_{p_i}(t) - \bar{v}_{p_i})^2} \cdot \sqrt{\sum_t (v_{p_j}(t) - \bar{v}_{p_j})^2}},$$

bu yerda $\max | \cdot |$ – guruhdagi eng uzoq ikki shaxs orasidagi masofa, ϵ_g – masofa chegarasi (odatda 2m), $corr()$ – tezlik vektorlarining Pirson korrelyatsiyasi, ρ_g – korrelyatsiya chegarasi. Birinchi shart guruhdagi barcha shaxslar bir-biriga yaqin ekanligini, ikkinchi shart ular bir xil yo'nalishda va tezlikda harakatlanayotganini tekshiradi. Ikkala shart bajarilsa, P to'plami guruh hisoblanadi.

Ushbu bo'limda kuzatish obyektlari uchun formal modellar keltirildi hamda ular orasidagi munosabatlar formallashtirildi. Bu obyektlar ustida 2.5-bo'limda aniqlangan funksional operatorlar (D, E, M, T, A) ishlaydi. Endi har bir operatorning matematik formulatsiyasi, algoritmlari va sifat mezonlari batafsil tavsiflanadi.

Ilg'or kuzatish tizimlari bir nechta o'zaro bog'liq vazifalarni ketma-ket yoki parallel bajaradi. Har bir vazifa o'z kirish va chiqishiga ega, o'z algoritmlariga asoslanadi va o'z sifat mezonlari bilan baholanadi. Ushbu bo'limda asosiy vazifalar mantiqiy ketma-ketlikda (tasvirdan yakuniy natijagacha) tavsiflanadi.



Obyektni aniqlash kuzatish tizimining birinchi va eng muhim bosqichi. Ushbu bosqichda kirish tasviridan barcha qiziqish obyektlari (shaxslar, transport vositalari, narsalar) topilishi va ularning joylashuvi aniqlanishi kerak. Bunda D aniqlash operatori berilgan $I \in R^{H \times W \times 3}$ tasvirni bounding boxlar, klasslar va ishonch darajalari to'plamiga akslantirishning formal ifodasi quyidagicha:

$$D: I \rightarrow (b_j, c_j, p_j)_{j=1}^{N_d}$$

bu yerda $b_j = (x, y, w, h)$ – bounding box (to'rtburchak paametrleri), $c_j \in C$ – klass, $p_j \in [0,1]$ – ishonch.

Obyektni aniqlash algoritmlari asosiy ikki guruhga bo'linadi: ikki bosqichli va bir bosqichli detektorlar.

Ikki bosqichli detektorlar avval tasvirdan qiziqarli hududlarni ajratib oladi, so'ngra har bir hududni alohida tasniflab aniqlashtiradi. Bu yondashuv yuqori aniqlik beradi, lekin qo'shimcha bosqich sababli nisbatan sekin. Faster R-CNN [5] ikki bosqichli detektorlarning eng mashhuri bo'lib, quyidagicha ishlaydi:

1-bosqichda RPN (Region Proposal Network, hududiy takliflar tarmog'i) ishlab, RPN tasvir ustida siljib, har bir nuqtada oldindan belgilangan anchor boxlar uchun ikki savol “bu yerda obyekt bormi (objectness score)” va “anchor boxni qanday sozlash kerak (bbox regression)” javobini bashorat qiladi:

$$p_{obj} = \sigma(W_{cls} \cdot f), \quad \Delta b = W_{reg} \cdot f,$$

bu yerda p_{obj} – obyekt mavjudligi ehtimoli (0 dan 1 gacha), σ – sigmoid funksiya, W_{cls} va W_{reg} – o'rganiladigan vaznlar matritsalarini, f – asosiy tarmoqdan ajratilgan xususiyatlar vektori, Δb – anchor box koordinatalariga qo'shiladigan tuzatish qiymatlari. RPN minglab potensial hududlarni taklif qiladi, ulardan eng yuqori ballilari ikkinchi bosqichga o'tkaziladi.

2-bosqichda har bir taklif qilingan hudud alohida qayta ishlanadi. Rol Pooling operatsiyasi (pool) orqali har bir hudud bir xil o'lchamga keltiriladi, so'ngra to'liq ulangan (fully connected) qatlamlar orqali klass aniq (classification) va koordinatalar yaxshilanadi (refinement):

$$p(c|r) = \text{softmax}(W_c \cdot \text{pool}(f, r)), \quad b_{final} = r + \Delta b'$$

bu yerda $p(c|r)$ – r hududida c klassining ehtimoli, softmax – ehtimolliklar taqsimotini hosil qiluvchi funksiya, W_c – klassifikatsiya vaznlari, $\text{pool}(f, r)$ – r hududidan olingan xususiyatlar vektori, b_{final} – aniq koordinatalar, $\Delta b'$ – ikkinchi bosqichda hisoblangan tuzatish. Ikki bosqichli detektorlar yuqori aniqlikda bo'lib, ular Tezkor R-CNN MS COCO datasetida 42.7% mAP ko'rsatadi. Biroq har bir hududni alohida qayta ishlash sekinlik olib keladi (odatda 5-10 FPS).

Bir bosqichli detektorlarda YOLO [6] tasvirni SxS katakchali panjaraga bo'lib va har bir katakchadan to'g'ridan-to'g'ri bashorat olinadi. Hududlarni taklif qilish va alohida qayta ishlash bosqichlari butunlay olib tashlanadi. Har bir katak o'z ichida B ta bounding box va ishonch darajasini hamda C ta klass ehtimolini bashorat qiladi:

$$y = (p_1, \dots, p_C, b_1, \dots, b_B, \text{conf}_1, \dots, \text{conf}_B),$$

bu yerda p_1, \dots, p_C – har bir klassga tegishlilik ehtimollari (masalan, p_1 – shaxs, p_2 – mashina), $b_k = (x, y, w, h)$ – B ta bounding box koordinatalari, conf_k – har bir box uchun ishonch darajasi.

Quyidagi jadval zamonaviy detection usullarining asosiy taqqoslash ko'rsatkichlarini solishtiradi:

1-jadval

Detection usullari taqqoslash

Usul	Backbone	mAP (COCO)	FPS	Parametrlar
Faster R-CNN	ResNet-101	42.7%	5-10	60M



YOLOv5-L	CSPDarknet	49.0%	45	47M
YOLOv8-L	CSPDarknet	53.9%	45	44M
DETR	ResNet-50	42.0%	28	41M

Obyekt aniqlangandan so'ng, uni boshqa obyektlardan farqlash uchun uning "portretini", ya'ni matematik tasvirini olish kerak. Bu jarayon embedding hosil qilish deyiladi. Bizda aniqlangan obyektning qirg'ilgan tasviri I_{crop} dan (ya'ni, kirish) d-o'lchamli embedding vektor $r \in R^d$ hosil qilish lozim. Buning maqsadi bir xil obyekt turli tasvirlarda o'xshash vektorga, turli obyektlar farqli vektorlarga ega bo'lishi kerak. Bu quyidagicha ifodalanadi:

$$E: I_{crop} \rightarrow e \in R^d, |e|_2 = 1.$$

Oddiy klassifikatsiya modellaridan farqli ravishda, embedding model test vaqtida o'qitishda ko'rmagan obyektlarni ham farqlashi kerak. Bu muammoni hal qilish uchun model konkret sinflarni emas, balki obyektlar orasidagi masofani o'rganishi kerak. Masofani o'rganishda Metric Learning yondashuvi qo'llaniladi. Maqsad shunday fazoni hosil qilishki, unda quyidagi shart bajarilsin:

$|e_i - e_j|_2 < |e_i - e_k|_2$ agar $y_i = y_j \neq y_k$, ya'ni, bir xil shaxs ($y_i = y_j$) tasvirlarining embeddinglari orasidagi masofa, turli shaxslar ($y_i \neq y_k$) embeddinglari orasidagi masofadan kichik bo'lishi kerak.

Hozirda bu maqsadga erishish uchun Triplet Loss, Softmax-based Loss, ArcFace Loss kabi turli muqobil loss funksiyalar taklif qilingan

Triplet Loss. Schroff [18] FaceNet da triplet loss ni taklif qilgan. Bu usulda o'qitish uchun uchliklar tanlanadi: anchor (a) – asosiy tasvir, positive (p) – bir xil shaxsning boshqa tasviri, negative (n) – boshqa shaxs tasviri. Binda anchor va positive yaqin, anchor va negative uzoq bo'lishi kerak:

$$L_{triplet} = \sum_{(a,p,n)} \max(0, |e_a - e_p|_2^2 - |e_a - e_n|_2^2 + \alpha),$$

bu yerda α – margin, uchliklarni tanlash (hard mining) qiyin va o'qitish sekin.

Softmax-based Loss usulda har bir shaxs alohida sinf deb qabul qilinadi va model qaysi sinfga tegishli ekanini bashorat qiladi:

$$L_{softmax} = -\log \frac{e^{W_y^T e + b_y}}{\sum_{j=1}^C e^{W_j^T e + b_j}},$$

bu yerda W_y – to'g'ri sinf vaznlari, C – sinflar soni. Biroq oddiy softmax sinflar orasidagi chegarani yetarlicha aniq ajratmaydi.

ArcFace Loss usulida Deng [9] angular margin qo'shishni taklif qilgan. Bunda embedding va sinf vaznlari orasidagi burchak hisoblanadi va to'g'ri sinfga kirish uchun qo'shimcha burchak (margin) talab qilinadi:

$$L_{arc} = -\log \frac{e^{s \cdot \cos(\theta_y + m)}}{e^{s \cdot \cos(\theta_y + m)} + \sum_{j \neq y} e^{s \cdot \cos \theta_j}}$$

Bu formula murakkab ko'rinadi, ammo g'oyasi oddiy: θ_y – to'g'ri sinf bilan burchak, m – "margin" (qo'shimcha masofa), s – masshtab. Margin qo'shish sinflar orasidagi chegarani kengaytiradi va model aniqroq farqlashni o'rganadi.

Embedding o'lchami tanlashda d parametri muhim hisoblanadi. Katta d ko'proq ma'lumot saqlaydi, ammo ko'proq xotira va hisoblash talab qiladi:

d	Xotira (vektor)	Tezlik	Aniqlik (LFW)
128	512 B	Tez	99.35%
256	1 KB	O'rta	99.65%
512	2 KB	O'rta	99.83%



2048	8 KB	Sekin	99.85%
------	------	-------	--------

Amalda $d=128$ ko'p holatlar uchun yetarli – aniqlik 0.5% farq, xotira 4 marta tejash.

Embedding operatori obyektning raqamli tasviri vektorni hosil qiladi. Bu vektorni bazadagi mavjud vektorlar bilan solishtirish orqali obyekt identifikatori aniqlanadi. Moslashtirish vazifasi aynan shu jarayonni amalga oshiradi, ya'ni yangi olingan embeddingni bazadagi embeddinglar bilan taqqoslaydi va eng mosini topadi. Bunda berilgan so'rov embeddingi e_{probe} va galereya $G = \{(id_i, e_i)\}_{i=1}^N$ uchun eng mos identifikatorni topish vazifasi quyidagicha formullashtiriladi:

$$M: (e_{probe}, G) \rightarrow (id^*, s^*),$$

bu yerda id^* – topilgan identifikator, s^* – o'xshashlik balli. Agar galereyadagi hech bir embedding yetarli darajada o'xshash bo'lmasa, obyekt "noma'lum" deb belgilanadi. Ikki vektor o'xshashligini aniqlash uchun turli metrikalar mavjud. Normallashtirilgan vektorlarda ($\|e\|=1$) kosinus o'xshashlik va Evklid masofasi eng keng qo'llaniladi.

Kosinus o'xshashlik: $sim(e_1, e_2) = \frac{e_1 \cdot e_2}{\|e_1\| \|e_2\|} = e_1 \cdot e_2$. Bunda qiymat $[-1, +1]$ gacha o'zgaradi: +1 aynan bir xil yo'nalish (maksimal o'xshash), 0 perpendikulyar (bog'lanish yo'q), -1 qarama-qarshi yo'nalishni bildiradi.

$$\text{Evklid masofasi: } dist(e_1, e_2) = |e_1 - e_2|_2 = \sqrt{2(1 - sim(e_1, e_2))}.$$

Normallashtirilgan vektorlar uchun bu ikki o'lchov matematik jihatdan ekvivalent, biri oshganda ikkinchisi kamayadi.

Galereyadagi eng o'xshash identifikatorni topish quyidagicha ifodalanadi:

$$id^* = \arg \max_{i \in [1, N]} sim(e_{probe}, e_i), s^* = sim(e_{probe}, e_{id^*}).$$

Qidiruv masalasida hisoblash murakkabligi muammosi bor. Oddiy (brute-force) qidirish har bir so'rov uchun barcha galereya elementlarini tekshirishni talab qiladi. Agar bazada juda katta embedding bo'lsa, har birini solishtirish juda sekin. Shuning uchun aniq qidirish o'rniga ANN (Approximate Nearest Neighbor – taqribiy eng yaqin qo'shni) algoritmlari qo'llaniladi. Malkov va Yashunin [19] HNSW algoritmini taklif qildilar. HNSW (Hierarchical Navigable Small World) algoritmi embeddinglarni maxsus ko'p qatlamli graf strukturasi saqlaydi. Bu graf asosidagi tuzilma $O(\log N)$ murakkablikda taqribiy eng yaqin qo'shni topadi.

Grafni qurish jarayoni:

1. Har bir yangi embedding uchun tasodifiy qatlam soni tanlash
2. Yuqori qatlamlardan boshlab, eng yaqin qo'shnilarni topib bog'lash
3. Quyi qatlamlarga tushilib, ko'proq bog'lanishlar qo'shish

Qidirish jarayoni:

1. Eng yuqori qatlamda kirish nuqtasidan (entry point) boshlash
2. Greedy search – har qadamda so'rovga eng yaqin qo'shni o'tish
3. Yaqinroq qo'shni topilmasa, quyi qatlamga tushib yanada aniqroq qidirish
4. Eng pastki qatlamda k ta eng yaqin qo'shni qaytarish

Bu yondashuv qidirish murakkabligini $O(N)$ dan $O(\log N)$ ga tushiradi. Million elementli bazada brute-force 100ms vaqt olsa, HNSW 1-2ms da natija beradi, natija 50-100 marta tezlashadi.

3-jadval

ANN algoritmlar taqqoslash ($N = 10^6, d = 128$)

Algoritm	Qurish vaqti	Qidirish vaqti	Recall@1	Xotira
Brute-force	–	100 ms	100%	512 MB
HNSW	5 min	1-2 ms	99.5%	1.5 GB
IVF-PQ	3 min	0.5 ms	95%	256 MB
Faiss-GPU	1 min	0.1 ms	99%	512 MB

Eng o'xshash element topilgandan so'ng, bu haqiqatan ham bir xil obyektmi yoki tasodifiy o'xshashlik b'ycha qaror qabul qilinadi. Buning uchun chegara qiymati (threshold) τ ishlatiladi:

$$qaror = \begin{cases} id^*, & \text{agar } s^* > \tau \\ noma'lum, & \text{aks holda} \end{cases}$$

τ qiymatini tanlash FAR (False Accept Rate) va FRR (False Reject Rate) o'rtasidagi muvozanatga bog'liq. Ya'ni, yuqori τ : past FAR, yuqori FRR (konservativ); past τ : yuqori FAR, past FRR (liberal). Amalda τ qiymati konkret vazifaning xavfsizlik talablariga qarab tanlanadi.

Aniqlash va identifikatsiya bitta kadrda boriladigan masala bo'lsa, obyektning kuzatish (tracking) esa video davomida izlash va izini qurish masala hisoblanadi. Video kuzatish tizimlarida alohida kadrlarda obyektlarni vaqt o'tishi bilan izchil kuzatib borish zarur. Ko'p obyektning kuzatish (Multi-Object Tracking, MOT) vazifasi aynan turli kadrlarda aniqlangan obyektlarni bir-biriga bog'lab, har bir obyekt uchun yagona traektoriya qurish muammosini hal qiladi.

Kadrlar ketma-ketligi $\{I_1, \dots, I_T\}$ va har bir kadrda aniqlashlar $\{B_t\}$ uchun traektoriyalar to'plamini qurish vazifasi quyidagich formullashtiriladi:

$$T: \{(B_t, E_t)\}_{t=1}^T \rightarrow \{\tau_i\}_{i=1}^M$$

bu yerda $B_t = \{b_1^t, \dots, b_{N_t}^t\}$ t vaqtdagi bounding boxlar, $E_t = \{e_1^t, \dots, e_{N_t}^t\}$ – ularning embeddingleri, τ_i – i -obyektning traektoriyasi.

Yuqoridagi vazifani yechish strategiyasi uchun asosiy qadamlar bajariladi, jumladan, harakat modelini qurish va pozitsiyani bashorat qilish (obyekt keyingi kadrda qayerda?), moslik darajasini hisoblash va optimal tayinlash (qaysi aniqlash qaysi traektoriyaga tegishli?), tashqi ko'rinish ma'lumotini qo'shish (faqat pozitsiya yetarli mi?), past sifatli aniqlashlar uchun ikki bosqichli strategiya (ByteTrack).

1. Harakat modeli va Kalman filtri. Obyektlar tasodifiy emas, balki fizik qonuniyatlar asosida traektoriyali harakatlanadi. Bu holat qonuniyatini modellashtirish uchun Kalman filtri keng qo'llaniladi. Har bir traektoriya uchun holat vektori quyidagicha aniqlanadi:

$$x = (u, v, s, r, \dot{u}, \dot{v}, \dot{s})^T,$$

bu yerda (u, v) – bounding box markazi koordinatalari, s – maydon, r – tomonlar nisbati (aspect ratio), nuqtali o'zgaruvchilar esa mos tezliklarni bildiradi.

Kalman filtri ikki bosqichda (bashorat va yangilash) ishlaydi. Bashorat bosqichida keyingi holatni oldingi holatdan hisoblash:

$$\hat{x}_{t|t-1} = F \cdot x_{t-1|t-1}; \hat{P}_{t|t-1} = F \cdot P_{t-1|t-1} \cdot F^T + Q,$$

bu yerda F – o'tish matritsasi, P – kovariansiya matritsasi, Q – jarayon shovqini.

Yangilash bosqichida, agar mos aniqlash topilsa, bashorat tuzatiladi:

$$K_t = \hat{P}_{t|t-1} H^T (H \hat{P}_{t|t-1} H^T + R)^{-1}; x_{t|t} = \hat{x}_{t|t-1} + K_t (z_t - H \hat{x}_{t|t-1})$$

bu yerda K_t – Kalman koeffitsienti, H – kuzatuv matritsasi, R – o'lchash shovqini, z_t – haqiqiy o'lchash (aniqlash).

2. Moslikni aniqlash va tayinlash masalasi. MOT ning markaziy muammosi ma'lumotlarni bog'lash (Data Association) bo'lib, unda ketma-ket kadrlardagi aniqlashlar mavjud traektoriyalar bilan to'g'ri bog'lash bajariladi. Faraz qilaylik, $t-1$ vaqtda M ta faol traektoriya mavjud va t vaqtda N ta yangi aniqlash olingan. Mazkur holatda har bir aniqlashni biror traektoriyaga tayinlash yoki yangi traektoriya sifatida boshlash vazifasi kelib chiqadi. Bunda har bir juftlik (i -traektoriya, j -aniqlash) uchun moslik darajasini baholovchi C_{ij} hisoblanadi. Har bir aniqlashni biror traektoriyaga tayinlashda umumiy farq minimal bo'lish lozim:

$$\pi^* = \arg \min_{\pi \in \Pi} \sum_{(i,j) \in \pi} C_{ij},$$



bu yerda Π – barcha mumkin 1:1 moslashuvlar to'plami, C_{ij} – i-traektoriya va j-aniqlash orasidagi farq (cost). Farq qanchalik past bo'lsa, moslik shunchalik ishonchli. Farqni hisoblashning eng sodda usuli bashorat qilingan va haqiqiy aniqlangan bounding box orasidagi kesishish ulushni hisoblashdir:

$$U(b_1, b_2) = \frac{|b_1 \cap b_2|}{|b_1 \cup b_2|}$$

IoU qiymatlari asosida farq matritsasi quriladi: $C_{ij} = 1 - IoU_{ij}$. Optimal tayinlashni topish uchun Venger algoritmi (Hungarian algorithm) qo'llaniladi. Bu algoritm $O(n^3)$ murakkablikda global optimal yechimni kafolatlaydi.

3. SORT algoritmi. Bewley va hammualiflari [12] yuqoridagi Kalman filtri va Venger algoritmini birlashtirib SORT (Simple Online and Realtime Tracking) algoritmini yaratdilar. Algoritm har bir kadrda quyidagi qadamlarni bajaradi:

- Mavjud traektoriyalar uchun Kalman filtri orqali pozitsiyani bashorat qilish
- Bashorat va yangi aniqlashlar orasida IoU hisoblab, farq matritsasini qurish
- Venger algoritmi orqali optimal tayinlashni topish
- Moslashgan traektoriyalarni yangilash, moslashmagan aniqlashlar uchun yangi traektoriya ochish.

SORT algoritmi sodda va tezkor (260+ FPS), u faqat fazoviy ma'lumotga tayanadi, ikki obyekt bir-biriga yaqinlashib o'tganda identifikatorlar almashishi (ID switch) mumkin.

4. Tashqi ko'rinish ma'lumotini qo'shish (DeepSORT). Faqat pozitsiyaga asoslangan moslik ba'zan yetarli emas. Masalan, ikki odam yaqindan o'tganda ularning bashorat qilingan pozitsiyalari bir-biriga yaqin bo'ladi va IoU asosida noto'g'ri tayinlash sodir bo'lishi mumkin. Wojke [13] bu muammoni hal qilish uchun tashqi ko'rinish (appearance) ma'lumotini kiritdi. Endi farq funksiyasi ikki komponentdan iborat:

$$C_{ij} = \lambda \cdot d_{motion}(i, j) + (1 - \lambda) \cdot d_{appearance}(i, j),$$

$$d_{motion}(i, j) = (d_j - \hat{d}_i)^T S_i^{-1} (d_j - \hat{d}_i), d_{appearance}(i, j) = 1 - e_i^T \cdot e_j,$$

bu yerda d_{motion} - harakat farqi Mahalanobis masofasi bilan; $d_{appearance}$ - tashqi ko'rinish farqi, u vektorlar orasidagi kosinus masofasi orqali hisoblanadi.

Har bir traektoriya uchun so'nggi L kadr (odatda L=100) davomidagi embeddinglar saqlanadi va yangi detection bilan solishtiriladi.

5. Past ishonchli aniqlashlar uchun ByteTrack. Zhang [14] mavjud usullarning muhim kamchiligini aniqladi, misol uchun ko'pchilik algoritmlarda faqat yuqori ishonchli aniqlashlar ($p > 0.5$) ishlatiladi, past ishonchlilar tashlab yuboradi.

Biroq, obyekt qisman yopilganda (okklyuziya) detektor past ishonch darajasini beradi. Bu ma'lumotni tashlash noto'g'ri, traektoriya uzilishi mumkin. ByteTrack ikki bosqichli strategiyani taklif qiladi:

- Yuqori ishonchli aniqlashlarni ($p > \theta_{high}$) bor traektoriyalar bilan bog'lash
- Moslashmagan traektoriyalar uchun past ishonchli aniqlashlar ($\theta_{low} < p < \theta_{high}$) bilan qayta urinish
- Natijada yangi traektoriyalar ochish, uzoq vaqt moslashmaganlarni o'chirish.

Bu yondashuv okklyuziya holatlarida yaxshiroq ishlaydi, ya'ni qisman yopilgan obyektga past ishonch beradi, ammo traektoriya davom etadi.

Sifat mezonlari. MOT tizimlarini baholash uchun bir nechta metrikalar ishlatiladi. MOTA (Multi-Object Tracking Accuracy) umumiy xatolarni o'lchaydi:

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t}$$

bu yerda FN – o'tkazib yuborilgan aniqlashlar, FP – noto'g'ri aniqlashlar, IDSW – identifikator almashuvlari, GT – haqiqiy obyektlar soni.

IDF1 (ID F1 Score) identifikatsiya izchilligini, HOTA (Higher Order Tracking Accuracy) esa detection va association sifatini alohida baholaydi.

Obyektni aniqlash, identifikatsiya qilish va kuzatib borish tizimning asosiy vazifalari hisoblanadi. Lekin shunday amaliy ilovalarda bu ko'rsatkichlari yetarli bo'lmasdan, balki, u nima qilayotganini ham baholash talab etiladi. Ayniqsa xavfsizlik tizimlarida obyekt holatini avtomatik aniqlash juda muhim.

Xatti-harakat tahlili obyektning traektoriyasi yoki video segmenti asosida obyektning harakatini klassifikatsiya qilishni nazarda tutib, bu vazifani quyidagicha formullashtiriladi:

$$A: \tau_i \rightarrow (action, prob)$$

bu yerda $action \in A = \{yurish, chopish, o'tirish, turish, yiqilish, jang, \dots\}$ – oldindan belgilangan harakatlar to'plami, $prob \in [0, 1]$ – klassifikatsiya ishonchliligi.

Harakatlar ikki guruhga bo'linadi: normal (yurish, o'tirish, turish) va anomal (yugurish, yiqilish, janjal). Anomal harakatlarni aniqlash tizimning asosiy maqsadlaridan biri.

Xatti-harakat tahlili uchun ikki yondashuv mavjud: RGB-based va Skeleton-based. RGB-based usullari (3D CNN, Two-Stream, I3D) to'g'ridan-to'g'ri video pikselleridan xususiyatlar ajratadi. Tran [24] tomonidan taklif qilingan I3D arxitekturasi 2D convolutionni vaqt o'lchamiga kengaytiradi:

$$f_{out} = \sigma(W * I_{t:t+k} + b)$$

bu yerda $I_{t:t+k}$ – ketma-ket k ta kadr, W – 3D convolution yadrosi (kernel). Kinetics datasetida 74.2% aniqlik ko'rsatildi. Biroq 3D convolution hisoblash jihatdan qimmat – real vaqt tizimlari uchun kam mos.

Skeleton-based usullar inson bo'g'imlarining koordinatalaridan foydalanib, bu ixcham, tez va tana harakatiga fokuslanadi. Zamonaviy kuzatish tizimlarida bu yondashuv keng qo'llanilib, unda inson skeleti graf sifatida modellashtiriladi:

$$G_{skeleton} = (V_{joints}, E_{bones})$$

bu yerda $V_{joints} = \{v_1, \dots, v_J\}$ – bo'g'imlar to'plami ($J=17 / 25$), E_{bones} – suyaklar (bo'g'imlarni bog'lovchi qirralar). Har bir bo'g'im vaqtga bog'liq koordinatalarga ega:

$$v_j(t) = (x_j(t), y_j(t), z_j(t), c_j(t))$$

bu yerda (x, y, z) – fazoviy koordinatalar, c – confidence (ko'rinish ishonchliligi).

Yan va hammualiflari [15] ST-GCN (Spatial-Temporal Graph Convolutional Network) arxitekturasini taklif qildilar. ST-GCN ikki o'lchamdagi bog'lanishlarni o'rganadi: spatial va temporal.

Spatial convolution – bir kadrdagi bo'g'imlar orasidagi bog'lanishlarni o'rganadi (masalan, qo'l va yelka orasidagi munosabat):

$$f_{out}^{spatial} = \sigma(\sum_k (A_k \odot M_k) \cdot f_{in} \cdot W_k),$$

bu yerda A_k – qo'shnilik matritsasi (qaysi bo'g'imlar anatomik jihatdan bog'langan, masalan tirsak-yelka, tizza-son), M_k – adaptiv e'tibor maskasi (qaysi bog'lanishlar muhimroq, masalan, "qo'l silkitish"da qo'l bo'g'imlari muhim, oyoq emas), W_k – convolution vaznlari, σ – aktivatsiya funksiyasi.

Temporal convolution – ketma-ket kadrlar orasidagi o'zgarishlarni o'rganadi. Bir xil bo'g'imning vaqt bo'yicha o'zgarishiga 1D convolution qo'llaniladi:

$$f_{out}^{temporal} = Conv1D(f_{out}^{spatial}, kernel_size = 9).$$

Kernel size = 9 degani 9 ta ketma-ket kadr (30 FPS da ~0.3 sekund) birgalikda tahlil qilinadi. ST-GCN da spatial va temporal convolution navbatma-navbat qo'llaniladi. Har bir blok avval fazoviy, keyin vaqtiy bog'lanishlarni o'rganadi. 9 ta bunday blok ketma-ket joylashtiriladi va oxirida harakat klassi aniqlanadi. NTU RGB+D datasetida 81.5% aniqlik ko'rsatildi.

Anomal xatti-harakat aniqlash. Yuqoridagi usullar oldindan belgilangan harakatlar to'plamida ishlaydi. Biroq real tizimlarda noma'lum, kutilmagan anomalialarni ham aniqlash zarur. Bunday holatlarda klassifikatsiya emas, rekonstruksiya yondashuvi qo'llaniladi. Autoencoder modeli normal xatti-harakatni qayta tiklashni o'rganadi:

$$L_{recon} = \|I - Dec(Enc(I))\|_2^2$$

bu yerda I – kirish kadri, $Enc(I)$ – latent vektor, $Dec(Enc(I))$ – qayta tiklangan kadr. L2 norma asl va qayta tiklangan tasvirlar orasidagi piksel bo'yicha farqni o'lchaydi. Model normal harakatlarni yaxshi qayta tiklashni o'rganadi, chunki faqat shunday ma'lumotlarni ko'rgan. Anomal harakat uchraganida model uni to'g'ri qayta tiklay olmaydi, ya'ni rekonstruksiya xatosi oshadi. Shunga ko'ra, kadrning anomal yoki normal ekanligini aniqlash quyidagi qoida orqali amalga oshiriladi:

$$anomaly(I) = 1[L_{recon}(I) > \theta_{anom}]$$

bu yerda $1[\cdot]$ – indikator funksiya (shart bajarilsa 1, aks holda 0), θ_{anom} – chegara qiymati. Agar rekonstruksiya xatosi chegaradan oshsa, kadr anomal deb belgilanadi.

Sultani va Shah [20] UCF-Crime datasetida bu yondashuvni sinovdan o'tkazdilar. Model hech qachon ko'rmagan turdagi anomaliyalarni (o'g'irlik, hujum) 75.4% AUC bilan aniqladi.

Ushbu bo'limda beshta asosiy vazifa alohida-alohida ko'rib chiqildi. Har bir vazifa uchun matematik formulatsiya, zamonaviy algoritmlar va sifat mezonlari keltirildi. Biroq real tizimda bu vazifalar alohida emas, balki yagona konveyer (pipeline) sifatida birgalikda ishlaydi. Keyingi bo'limda vazifalarning integratsiyasi, xatolar tarqalishi va umumiy optimallashtirish masalalari ko'rib chiqiladi.

Oldingi bo'limda intellektual kuzatish tizimining beshta asosiy vazifasi – aniqlash (D), embedding (E), moslashtirish (M), kuzatib borish (T) va xatti-harakat tahlili (A) – alohida-alohida ko'rib chiqildi. Har bir vazifa o'z kirish va chiqishiga ega, o'z algoritmlariga asoslanadi. Biroq real tizimda bu vazifalar alohida emas, balki yagona konveyer (pipeline) sifatida birgalikda ishlaydi. Ushbu bo'limda vazifalar orasidagi bog'lanishlar, ma'lumot oqimi va umumiy optimallashtirish masalalari ko'rib chiqiladi.

Intellektual kuzatish tizimida vazifalar qat'iy ketma-ketlikda bajariladi – har bir vazifaning chiqishi keyingi vazifaning kirishi bo'ladi. Bu ketma-ketlikni konveyer (pipeline) deb atalad. Bitta kadr I_t uchun ma'lumot oqimi quyidagicha ifodalanadi:

$$I_t \xrightarrow{D} B_t \xrightarrow{E} E_t \xrightarrow{M} ID_t \xrightarrow{T} T_t \xrightarrow{A} A_t$$

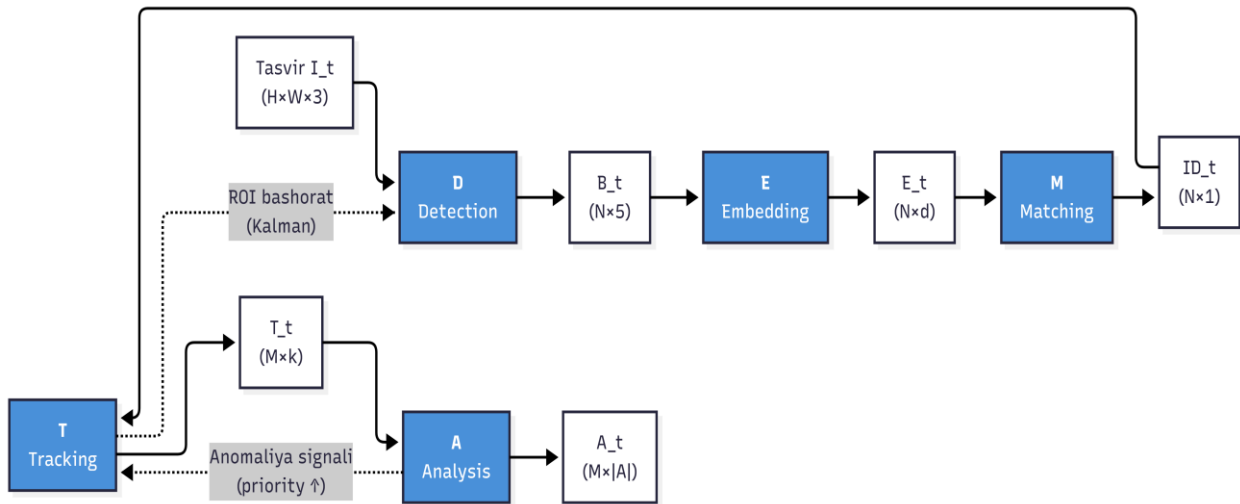
bu yerda I_t – t vaqtdagi kirish tasviri, B_t – aniqlangan obyektlar to'plami (bounding boxlar), E_t – embedding vektorlar to'plami, ID_t – identifikatorlar, T_t – yangilangan traektoriyalar, A_t – aniqlangan harakatlar.

Har bir bosqichda ma'lumot hajmi va shakli o'zgaradi. Tasvirdan traektoriyagacha bo'lgan o'zgarishni kuzatish mumkin:

$$\underbrace{I_t}_{H \times W \times 3} \rightarrow \underbrace{B_t}_{N \times 5} \rightarrow \underbrace{E_t}_{N \times d} \rightarrow \underbrace{ID_t}_{N \times 1} \rightarrow \underbrace{T_t}_{M \times k} \rightarrow \underbrace{A_t}_{M \times |A|}$$

bu yerda $H \times W \times 3$ – tasvir o'lchami (masalan, $1920 \times 1080 \times 3 \approx 6$ MB), N – aniqlangan obyektlar soni, d – embedding o'lchami (128/512), M – faol traektoriyalar soni, k – traektoriya parametrlari, $|A|$ – harakatlar klasslar soni.

Vazifalar konveyerining umumiy sxemasi 2-rasmda keltirilgan. Sxemada asosiy ma'lumot oqimi bilan birga qayta aloqa (feedback) yo'nalishlari ham ko'rsatilgan – tracking operatori detection ga ROI bashorat signalini, analysis operatori tracking ga anomaliya signalini yuboradi.



2-

rasm. Vazifalar konveyeri va qayta aloqa mexanizmlari

Konveyerning muhim xususiyati shundaki, har bir bosqich oldingi bosqich natijasiga to'liq bog'liq. Agar D operatori obyektни o'tkazib yuborsa, keyingi bosqichlar (E, M, T, A) bu obyekt uchun umuman ishlamaydi. Bu xususiyat xatolar tarqalishi (error propagation) muammosini keltirib chiqaradi.

Konveyer arxitekturasida dastlabki bosqichdagi xato keyingi bosqichlarga tarqaladi va ko'payadi. Bu muammoni formal tavsiflash mumkin. Har bir operatorning xato ehtimoli quyidagicha belgilanadi:

$$\epsilon_D = P(\text{detection xato}), \epsilon_E = P(\text{embedding xato}), \dots$$

Agar operatorlar mustaqil bo'lsa, yakuniy to'g'ri natija ehtimoli:

$$P(\text{to'g'ri natija}) = (1 - \epsilon_D)(1 - \epsilon_E)(1 - \epsilon_M)(1 - \epsilon_T)(1 - \epsilon_A)$$

Misol: har bir operator 95% aniqlikda ishlasa ($\epsilon = 0.05$), $P=0.774$. Ya'ni 5 ta operator birgalikda faqat 77.4% aniqlik beradi. Bu natija konveyerning zaif tomonini ko'rsatib, har bir bosqich imkon qadar yuqori aniqlikda ishlashi zarur.

Xatolar tarqalishini kamaytirish uchun bir nechta yondashuvlar mavjud: 1) har bir operatorni alohida optimallashtirish, ya'ni eng yaxshi detection, eng yaxshi embedding modellarni tanlash; 2) qayta aloqa (feedback) mexanizmlari, ya'ni keyingi bosqichlardan oldingi bosqichlarga signal yuborish; 3) end-to-end o'qitish, ya'ni barcha operatorlarni birgalikda optimallashtirish.

An'anaviy konveyerda ma'lumot faqat oldinga oqadi. Biroq samarali tizimda keyingi bosqichlar oldingi bosqichlarga foydali signal yuborishi mumkin. Bu qayta aloqa (feedback) deb ataladi.

Kuzatishdan aniqlashga qayta aloqa. Tracking operatori T oldingi kadrlar asosida obyektning keyingi pozitsiyasini bashorat qiladi. Bu bashorat D operatoriga "qayerga qarash kerak" signalini beradi:

$$\hat{x}_t = F \cdot x_{t-1} \Rightarrow ROI_t = \text{expand}(\hat{x}_t, \delta),$$

bu yerda F – Kalman filtri o'tish matritsasi (4.4-bo'limda), ROI_t – qidiruv hududi (Region of Interest), δ – kengaytirish chegarasi. Detektor butun tasvirni emas, faqat bashorat qilingan hududni qayta ishlashi mumkin. Bu tezlikni oshiradi va soxta aniqlashlarni (false positive) kamaytiradi.

Tahlildan kuzatishga qayta aloqa. Xatti-harakat tahlili A anomaliya aniqlasa, bu signal T operatoriga yuboriladi:

$$\text{anomaly}(o_i) = 1 \Rightarrow \text{priority}(o_i) \uparrow, \theta_{lost}(o_i) \uparrow$$

ya'ni anomal obyekt uchun kuzatish prioriteti oshadi va yo'qolish chegarasi (θ_{lost}) kengayadi, tizim bu obyektни yo'qotmaslikka harakat qiladi.

Har bir operator o'z maqsad (loss) funksiyasiga ega. Butun tizimni optimallashtirish uchun bu funksiyalarni birlashtirish zarur. Umumiy maqsad funksiyasi quyidagicha ifodalanadi:

$$L_{total} = \lambda_D L_D + \lambda_E L_E + \lambda_T L_T + \lambda_A L_A$$

bu yerda L_D – detection loss (focal loss, box regression), L_E – embedding loss (ArcFace), L_T – tracking loss (association sifati), L_A – xatti-harakat loss (cross-entropy), $\lambda_D, \lambda_E, \lambda_T, \lambda_A$ – muvozanat koeffitsientlari. Muvozanat koeffitsientlari λ har bir vazifaning ahamiyatini belgilaydi. Agar xavfsizlik tizimida identifikatsiya muhimroq bo'lsa, λ_E oshiriladi. Agar xatti-harakat tahlili muhimroq bo'lsa, λ_A oshiriladi. Amalda bu koeffitsientlar validatsiya to'plamida tanlanadi.

Biroq yuqoridagi yondashuv har bir operatorni alohida o'qitishni nazarda tutadi. Zamonaviy tadqiqotlar barcha operatorlarni birgalikda, yagona tarmoq sifatida o'qitishni taklif qiladi.

An'anaviy yondashuvda har bir operator alohida o'qitiladi: avval detection modeli, keyin embedding modeli va hokazo. Bu yondashuv sodda, lekin operatorlar orasidagi o'zaro ta'sirni hisobga olmaydi.

End-to-end (boshidan oxirigacha) o'qitishda barcha operatorlar yagona tarmoq sifatida birlashtiriladi va gradient barcha qatlamlar orqali o'tadi:

$$\theta^* = \arg \min_{\theta} L_{total}(\theta_D, \theta_E, \theta_T, \theta_A)$$

bu yerda $\theta = \{\theta_D, \theta_E, \theta_T, \theta_A\}$ – barcha operatorlar parametrlari.

Wang va hammualiflari [22] JDE (Joint Detection and Embedding) arxitekturasida detection va embedding ni bitta tarmoqda birlashtirishdi:

$$L_{JDE} = L_D + \alpha L_E$$

bu yerda α – muvozanat koeffitsienti. Bitta asosiy tarmoqdan (backbone) ikkita bosh (head) chiqadi: biri detection uchun, biri embedding uchun. Bu yondashuv inference (xulosa chiqarish) vaqtini 2 marta qisqartirdi, alohida detection va embedding o'rniga bitta oldinga o'tish (forward pass) yetarli.

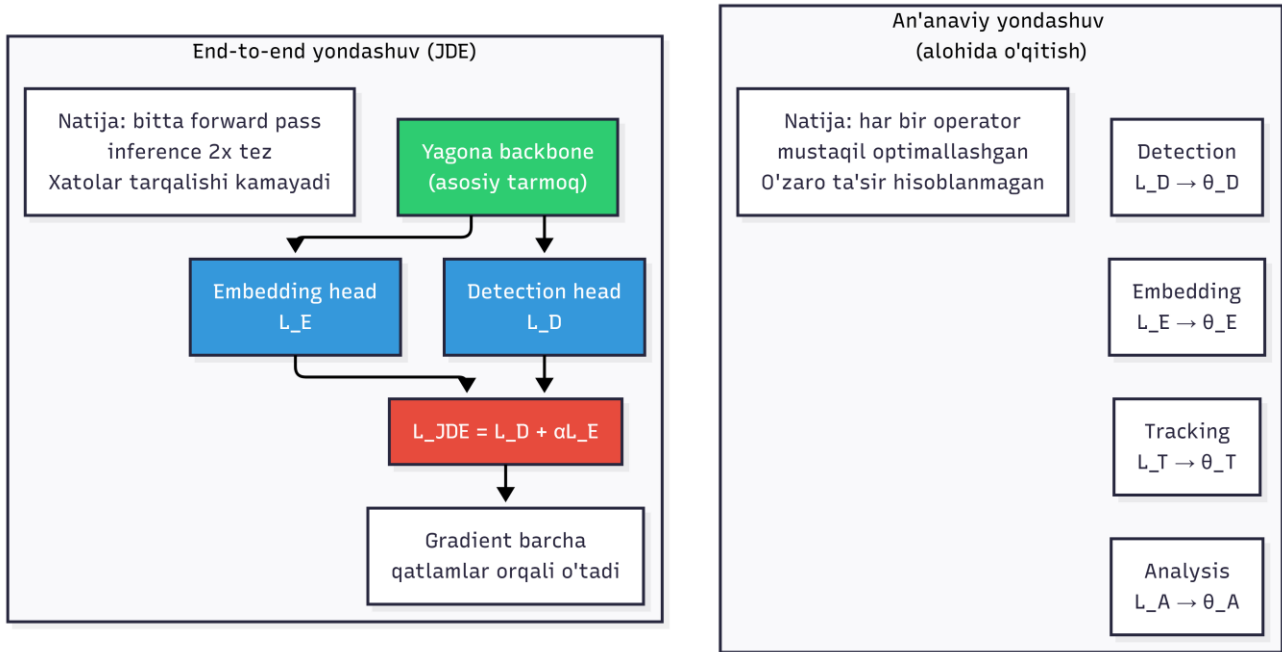
An'anaviy va end-to-end yondashuvlar orasidagi farq 3-rasmda ko'rsatilgan. An'anaviy yondashuvda har bir operator o'z loss funksiyasi bilan mustaqil o'qitiladi, operatorlar orasidagi o'zaro ta'sir hisobga olinmaydi. End-to-end yondashuvda esa yagona backbone tarmoqdan ikki bosh (head) chiqadi va gradient barcha qatlamlar orqali o'tib, operatorlar birgalikda optimallasadi.

End-to-end o'qitishda operatorlar birgalikda optimallasadi, xatolar tarqalishi kamayadi, inference tezligi oshadi. Lekin, o'qitish murakkablashadi, ko'proq xotira talab qilinadi, debug qilish qiyinlashadi.

Butun tizim sifatini baholash uchun alohida operatorlar mezonlari yetarli emas. Tizim darajasidagi mezonlar quyidagilarni o'z ichiga oladi:

- Aniqlash to'liqligi: $Recall_{system} = \frac{|o:detected(o) \wedge identified(o)|}{|O_{total}|}$,
- Identifikatsiya aniqligi: $Precision_{ID} = \frac{|o:ID(o)=ID_{true}(o)|}{|o:detected(o)|}$,
- Traektoriya sifati: $Q_{avg} = \frac{1}{M} \sum_{i=1}^M Q(\tau_i)$, M – traektoriyalar soni;
- Real vaqt ko'rsatkichi: $Latency_{total} = t_D + t_E + t_M + t_T + t_A \leq \frac{1}{FPS_{required}}$.

Masalan, 30 FPS talab qilinsa, umumiy latency 33 ms dan oshmasligi kerak.



3-rasm. An'anaviy va end-to-end (JDE) yondashuvlar taqqoslash sxemasi

Xulosa

Mavjud tadqiqotlar intellektual kuzatish tizimlarining alohida vazifalarini – aniqlash, identifikatsiya, kuzatish – mustaqil masalalar sifatida ko'rib chiqadi. Bibliometrik tahlil natijasi buni tasdiqladi: 1,480 dan ortiq tadqiqotning 87% bitta vazifaga qaratilgan, 13% ikkitasini birlashtirgan (masalan, JDE [22] da detection va embedding), biroq barcha beshta vazifani yagona matematik framework da formal tavsiflagan ish topilmadi. Ushbu maqolada aynan shu bo'shliqni to'ldirish maqsadida belgilangan hududlarda obyektlarni maqsadli kuzatish va boshqarishning matematik asoslari ishlab chiqildi.

Tadqiqotning asosiy natijalari quyidagilardan iborat:

1. Kuzatish tizimining umumiy matematik modeli $S = (O, H, V, T, \Phi)$ taklif etildi. Mavjud yondashuvlardan farqli o'laroq, ushbu model barcha beshta komponentni – kuzatish obyektlari (O), nazorat hududlari (H), kuzatish vositalari (V), vaqt oralig'i (T) va funksional operatorlar (Φ) – yagona framework ichida birlashtiradi. Hudud modeli H ichidagi o'tish matritsasi R to'g'ridan-to'g'ri anomaliya aniqlash operatoriga bog'langan – bu mavjud ishlarda formallashtirilmagan.

2. Kuzatish obyektlari uchun formal modellar ishlab chiqildi. Shaxs obyekti uchun uch darajali atributlar tizimi (biometrik, tashqi ko'rinish, xatti-harakat) aniqlandi. Obyektlar orasidagi munosabatlar matematik shaklda ifodalandi.

3. Beshta asosiy vazifa (aniqlash, embedding, moslashtirish, kuzatib borish va xatti-harakat tahlili) yagona notatsiyada matematik formulatsiya bilan tavsiflandi. Har bir vazifa uchun kirish/chiqish, maqsad funksiyasi va sifat mezonlari aniqlandi. Zamonaviy algoritmlar qiyosiy tahlil qilindi:

- YOLOv8 – 53.9% mAP (MS COCO),
- ArcFace – 99.83% (LFW),
- ByteTrack – 80.3 MOTA (MOT17),
- ST-GCN – 81.5% (NTU RGB+D).

4. Vazifalarning yagona konveyer sifatida integratsiyasi tahlil qilindi. Xatolar tarqalishi muammosi formal ko'rsatildi, besh bosqichli konveyerda har bir operator 95% aniqlikda ishlasa ham yakuniy natija 77.4% ga tushadi. Bu muammoni yumshatish uchun ikki yo'nalishli qayta aloqa mexanizmlari (tracking → detection, analysis → tracking)



formallashtirildi va end-to-end o'qitish (JDE arxitekturasi) inference vaqtini 2 marta qisqartirishi ko'rsatildi.

Tadqiqotning ilmiy ahamiyati sifatida taklif etilgan $S = (O, H, V, T, \Phi)$ model (1) yangi algoritmlarni ishlab chiqishda nazariy asos sifatida xizmat qiladi; (2) mavjud tizimlarni tahlil qilish va taqqoslash uchun yagona mezon beradi; (3) turli hududlar uchun moslashtirish imkonini yaratadi.

Ushbu maqolada statik kamera tarmoqlari ko'rib chiqildi. Mobil kameralar (dronlar, shaxsiy qurilmalar) uchun model kengaytirish alohida tadqiqot talab qiladi. 3D rekonstruksiya va ko'p ko'rinishli geometriya integratsiyasi kuzatish aniqligini oshirishi mumkin. Nazoratli va nazoratsiz usullar katta hajmli belgilanmagan ma'lumotlardan foydalanish imkonini beradi. Hisoblash infratuzilmasi (edge-fog-cloud arxitektura), real vaqt optimizatsiyasi va resurs boshqaruvi masalalari kelgusi tadqiqotlarda ko'rib chiqiladi.

Foydalanilgan adabiyotlar

- [1] United Nations. World Urbanization Prospects: The 2018 Revision // Department of Economic and Social Affairs. – 2018.
- [2] Gill M., Spriggs A. Assessing the impact of CCTV // Home Office Research Study. – 2005. – № 292.
- [3] Tickner A. H., Poulton E. C. Monitoring up to 16 synthetic television pictures // Ergonomics. – 1973. – T. 16. – № 4. – C. 381–401.
- [4] Girshick R., Donahue J., Darrell T., Malik J. Rich feature hierarchies for accurate object detection // Proceedings of CVPR. – 2014. – C. 580–587.
- [5] Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards real-time object detection // Advances in Neural Information Processing Systems (NeurIPS). – 2015. – C. 91–99.
- [6] Redmon J., Divvala S., Girshick R., Farhadi A. You only look once: Unified real-time object detection // Proceedings of CVPR. – 2016. – C. 779–788.
- [7] Jocher G., Chaurasia A., Qiu J. Ultralytics YOLOv8 // GitHub repository. – 2023. – URL: <https://github.com/ultralytics/ultralytics>
- [8] Taigman Y., Yang M., Ranzato M., Wolf L. DeepFace: Closing the gap to human-level performance // Proceedings of CVPR. – 2014. – C. 1701–1708.
- [9] Deng J., Guo J., Xue N., Zafeiriou S. ArcFace: Additive angular margin loss for deep face recognition // Proceedings of CVPR. – 2019. – C. 4690–4699.
- [10] He S., Luo H., Wang P., et al. TransReID: Transformer-based object re-identification // Proceedings of ICCV. – 2021. – C. 15013–15022.
- [11] Ye M., Shen J., Lin G., et al. Deep learning for person re-identification: A survey and outlook // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2021. – T. 44. – № 6. – C. 2872–2893.
- [12] Bewley A., Ge Z., Ott L., et al. Simple online and realtime tracking // Proceedings of ICIP. – 2016. – C. 3464–3468.
- [13] Wojke N., Bewley A., Paulus D. Simple online and realtime tracking with a deep association metric // Proceedings of ICIP. – 2017. – C. 3645–3649.
- [14] Zhang Y., Sun P., Jiang Y., et al. ByteTrack: Multi-object tracking by associating every detection box // Proceedings of ECCV. – 2022. – C. 1–21.
- [15] Yan S., Xiong Y., Lin D. Spatial temporal graph convolutional networks for skeleton-based action recognition // Proceedings of AAAI. – 2018. – C. 7444–7452.
- [16] Liu X., Liu W., Ma H., Fu H. Large-scale vehicle re-identification // Proceedings of ICME. – 2016. – C. 1–6.
- [17] Schroff F., Kalenichenko D., Philbin J. FaceNet: A unified embedding for face recognition and clustering // Proceedings of CVPR. – 2015. – C. 815–823.



[18] Malkov Y. A., Yashunin D. A. Efficient and robust approximate nearest neighbor search using HNSW graphs // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2018. – T. 42. – № 4. – С. 824–836.

[19] Sultani W., Chen C., Shah M. Real-world anomaly detection in surveillance videos // Proceedings of CVPR. – 2018. – С. 6479–6488.

[20] Sun Y., Cheng C., Zhang Y., et al. Circle loss: A unified perspective of pair similarity optimization // Proceedings of CVPR. – 2020. – С. 6398–6407.

[21] Wang Z., Zheng L., Liu Y., et al. Towards real-time multi-object tracking // Proceedings of ECCV. – 2020. – С. 107–122.

[22] Ge W., Collins R. T., Ruber R. B. Vision-based analysis of small groups in pedestrian crowds // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2012. – T. 34. – № 5. – С. 1003–1016.

[23] Carreira J., Zisserman A. Quo vadis, action recognition? A new model and the Kinetics dataset // Proceedings of CVPR. – 2017. – С. 6299–6308.