GHR-VLM: Making Zero-Shot Transit Video Analytics Realizable with Grounded Hybrid Reasoning

Kaicong Huang
huangk10@rpi.edu
Rensselaer Polytechnic Institute
Troy, NY, USA

Weiheng Oh
ohw@rpi.edu
Rensselaer Polytechnic Institute
Troy, NY, USA

Jack M. Reilly
REILLJ2@rpi.edu
Rensselaer Polytechnic Institute
Troy, NY, USA

Thomas Guggisberg
thomas@cdta.org
Capital District Transportation Authority
Albany, NY, USA

Ruimin Ke*
ker@rpi.edu
Rensselaer Polytechnic Institute
Troy, NY, USA

arXiv:2607.13569v2 [cs.CV] 20 Jul 2026

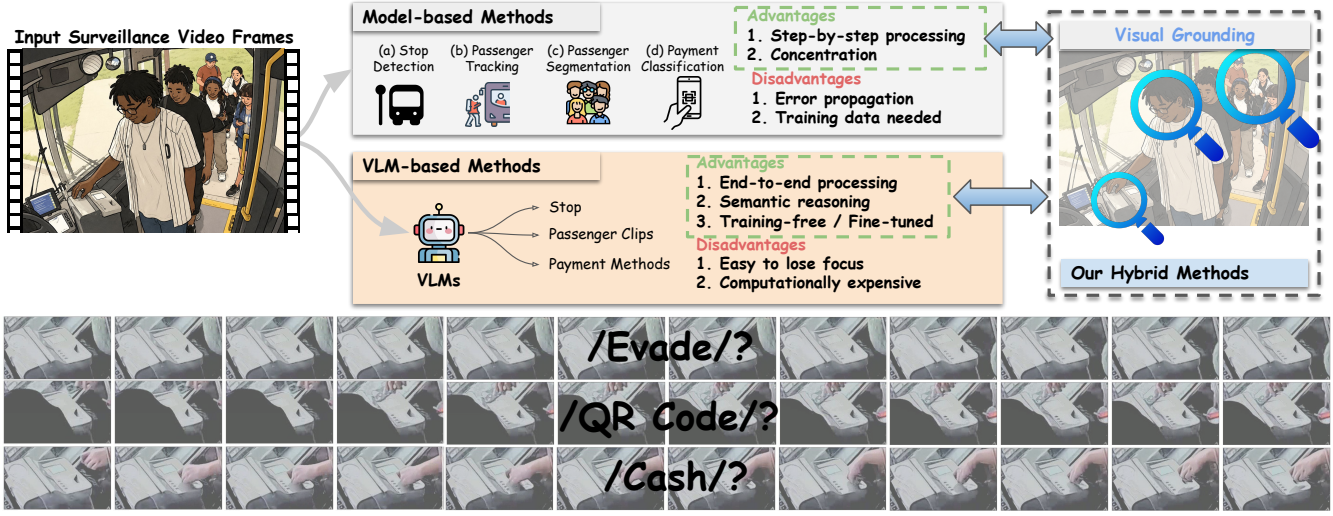

Figure 1: Comparison of model-based, VLM-based, and our grounded hybrid approaches for transit video analytics. Model-based pipelines provide explicit visual grounding but require task-specific training and suffer from error propagation. VLM-based methods offer open-vocabulary reasoning but may lose focus in long videos and incur high inference costs. Our hybrid approach integrates grounded stepwise processing with selective VLM reasoning, enabling zero-shot transit video analytics.

Abstract

Transit video understanding can provide valuable fine-grained data that conventional passenger counters and fare systems cannot capture. However, supervised video models require task-specific annotations, while applying vision-language models (VLMs) directly to long onboard videos is unreliable and costly. To leverage the complementary strengths of both approaches, we propose GHR-VLM, a visual grounded hybrid reasoning framework for zero-shot transit-bus video analytics. It is motivated by the observation that explicit visual grounding can improve VLM reasoning by converting long surveillance streams into compact, passenger-centered spatiotemporal evidence. Specifically, we propose an edge-cloud design in which a lightweight edge-based monitor continuously tracks door status and segments passenger clips. A backend VLM then identifies boarding passengers and classifies payment behavior through a two-stage coarse-to-fine refinement of spatiotemporal evidence. By invoking the VLM only on grounded passenger clips and contact sheets, GHR-VLM reduces cloud inference, avoids payment-specific training data, and supplies the localized evidence that VLMs otherwise struggle to identify. Evaluation on 486 minutes of real-world bus surveillance video demonstrates the potential of grounded

*Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference'17, Washington, DC, USA

© 2026 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

edge-cloud reasoning for passenger-level payment analytics while highlighting the challenges posed by degraded video conditions.

CCS Concepts

• **Computing methodologies** → **Artificial intelligence**; **Machine learning**; **Planning and scheduling**.

Keywords

transit video analytics, vision-language models, visual grounding, edge-cloud collaboration, zero-shot recognition

ACM Reference Format:

Kaicong Huang, Weiheng Oh, Jack M. Reilly, Thomas Guggisberg, and Ruimin Ke. 2026. GHR-VLM: Making Zero-Shot Transit Video Analytics Realizable with Grounded Hybrid Reasoning. In . ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Transit agencies require fine-grained operational data for service planning, crowding management, revenue auditing, and passenger-experience assessment [10]. Existing passenger counters, fare logs, and inspection records provide only partial evidence and cannot associate each stop event with individual passenger activities [7, 8, 12]. Automated payment analytics must therefore identify boarding and alighting passengers, localize payment events, recognize payment methods, and detect unpaid boarding.

This problem involves vehicle-state detection, passenger tracking, temporal segmentation, activity understanding, and fine-grained hand-object recognition. Existing fare-evasion methods learn specific behaviors from annotated keypoints or action examples [27], while more general video models acquire spatiotemporal representations through supervised training [1, 4, 9, 11, 26, 30]. Their reliance on task-specific labeled data, however, limits their applicability to onboard payment analytics, where payment behaviors are **diverse, sparsely observed, and costly to annotate**. We therefore formulate the problem as a **zero-shot fine-grained spatiotemporal task**, in which brief farebox interactions must be localized and associated with the correct passenger in low-resolution, cluttered video.

Video vision-language models (VLMs) offer a promising language-driven interface for such difficult-to-annotate events. Image- and video-language models have demonstrated open-vocabulary recognition, visual grounding, instruction following, multimodal reasoning, and temporal understanding [2, 3, 15, 17, 18, 20, 25, 28, 31]. However, directly prompting VLMs on long onboard videos is unreliable and expensive. Spatially, blur, poor illumination, low resolution, and occlusion obscure the localized cues that distinguish payment methods. Temporally, payment actions occupy only a small fraction of the stream and require precise segmentation. Fleet-scale cloud inference also conflicts with edge constraints on latency, bandwidth, and computation [24].

These limitations motivate us to combine VLM reasoning with explicit spatiotemporal grounding. Detectors, segmentation models, and trackers can localize passengers, associate trajectories, and extract relevant intervals [6, 29, 32], while open-vocabulary grounding and promptable segmentation reduce the annotation effort required for deployment [14, 16, 19, 21–23]. Together, these components can transform long surveillance streams into compact, passenger-centered evidence for VLM reasoning.

We propose GHR-VLM, a Visual Grounded Hybrid Reasoning framework for zero-shot transit-bus video analytics. Lightweight edge modules identify the front-door and payment regions, extract stop intervals from door states, and detect, track, and separate passengers within those intervals. A backend VLM then distinguishes boarding passengers from alighting or already-onboard passengers and classifies payment behavior through a two-stage coarse-to-fine refinement of spatiotemporal farebox evidence. By invoking the VLM only on grounded clips and contact sheets, GHR-VLM reduces cloud inference, avoids task-specific payment training data, and supplies the precise evidence that VLMs otherwise struggle to localize.

This paper makes the following contributions:

- We introduce a grounded hybrid reasoning framework for zero-shot transit video analytics, in which model-based perception provides structured spatiotemporal evidence to guide the open-vocabulary semantic reasoning of VLMs.
- We develop an end-to-end bus payment data collection pipeline that transforms a single onboard surveillance stream into structured stop-level, passenger-level, and payment-level events.
- We instantiate an edge-cloud collaborative architecture designed for resource-constrained in-vehicle deployment.

2 Transit Data Collection

Transit data collection is typically divided into passenger counting, transaction logging, and inspection planning. Learned counters improve boarding and alighting estimates but remain focused on aggregate counts [12]. Fare-card and farebox records provide transaction data, yet they contain blind spots such as unrecorded evasion and incomplete deployment, motivating their fusion with passenger counts [8]. Operations research further addresses fare evasion through network-level inspection and deterrence models [7]. However, none of these approaches reconstructs a passenger-level visual chain from boarding to payment. Vision-based methods address parts of this problem through supervised fare-evasion and action recognition [1, 4, 9, 11, 26, 27, 30]. Although effective with labeled data and stable scene geometry, they require task-specific training and usually predict only narrow fraud categories rather than complete payment events. Moreover, onboard buses are less constrained than stations or gates. Passengers enter in groups, occlude one another, use diverse payment media, and interact with small farebox components under poor lighting. GHR-VLM therefore formulates transit data collection as grounded video understanding rather than isolated counting or fare-evasion classification.

3 Methodology

Given a continuous onboard video $\mathcal{V} = \{I_t \mid 0 \le t \le T\}$, GHR-VLM produces passenger-level transit records

$$\mathcal{R} = \left\{(S_k, P_{k,j}, \hat{a}_{k,j}, \hat{y}_{k,j})\right\}_{k,j} \tag{1}$$

where I_t is the frame at timestamp t, T is the video duration, S_k is the k-th stop interval, $P_{k,j}$ is the j-th passenger clip within that stop, $\hat{a}_{k,j}$ is the passenger direction, and $\hat{y}_{k,j}$ is the payment label. The label space is $\mathcal{Y} = \{\text{QR}, \text{cash}, \text{tap}, \text{swipe}, \text{evade}\}$. Lightweight and promptable vision modules continuously localize relevant evidence, while VLM inference is reserved for compact grounded inputs. Figure 2 shows the four stages described below.

3.1 Door-Grounded Stop Filtering

Most frames in a continuous trip contain no passenger exchange, so GHR-VLM uses the front door to identify likely stops. SAM 3 [5] localizes the front entrance in the first processed frame and returns a normalized box B_d, which is cached for the full video under the fixed-camera setting. The box is expanded by 20% to retain the doorway context, while pixels outside it are masked to suppress irrelevant motion.

The VLM analyzes the cropped stream in overlapping 10-second windows, each represented by 12 uniformly sampled frames. It predicts *no change*, *closed-to-open*, or *open-to-closed*, and identifies the first frame showing the new state. A second VLM call verifies each candidate within a five-second window on either side of its timestamp, and duplicate transitions within three seconds are merged. The verified pins are $\mathcal{P}_d = \{(\hat{t}_\ell, \hat{z}_\ell)\}_{\ell=1}^{L}$, where L is the number of retained pins, $\hat{t}_\ell$ is a timestamp, and $\hat{z}_\ell \in \{c \to o, o \to c\}$ is the transition type.

Let t_k^o denote the timestamp of the k-th opening pin, and let C_k contain all verified closing timestamps after t_k^o. The matched closing time is

$$t_k^c = \min C_k \tag{2}$$

When C_k is empty, the processing endpoint is used as t_k^c. The resulting stop interval is

$$S_k = [t_k^o, t_k^c] \tag{3}$$

Only frames within these intervals enter passenger tracking. We also provide a lightweight alternative that applies a fine-tuned YOLO11 [13] to each grounded door crop.

3.2 Rule-Based Passenger Tracking and Temporal Segmentation

Each stop interval S_k defines the temporal range for passenger tracking. The system then processes alternate frames and enhances interior visibility through adaptive contrast, shadow amplification, and highlight suppression. SAM 3 tracks instances prompted as "passenger" in 16-frame batches, while boxes covering less than 5% of the image or wider than tall are discarded.

Batching reduces edge memory usage but can reset identities across batches. We therefore match each detection in the first frame of a new batch to the best unmatched detection in the final frame of the previous batch. Their overlap is

$$\text{IoU}(B, B') = \frac{|B \cap B'|}{|B \cup B'|} \tag{4}$$

where B is a boundary box from the preceding batch and B' is a boundary box from the new batch. A match is accepted when the IoU exceeds $\theta_{\text{I}} = 0.20$. The matched local identity inherits the corresponding global identity.

To identify the primary passenger in each frame, we exploit the fixed camera geometry as a projective prior, assuming that the passenger closest to the fare area is the target. Let $\mathcal{D}_{k,i}$ be the valid tracked identities visible in sampled frame i of stop k, and let $B_{i,j}$ be the box of identity j. The main identity is selected by

$$m_i = \arg\max_{j \in \mathcal{D}_{k,i}} \text{area}(B_{i,j}) \tag{5}$$

The index $i \in \{1, \ldots, N_k\}$ identifies one of the N_k sampled frames. The value m_i is set to -1 when $\mathcal{D}_{k,i}$ is empty.

To mitigate brief identity switches under poor visual conditions, we apply a two-step repair that converts the raw sequence m_i into $\tilde{\mathbf{m}}_k$. First, a one-frame identity differing from both neighbors is replaced by the preceding identity. Second, a valid run of up to three frames is replaced by the preceding identity, while a missing run of the same length is filled only when its neighboring identities agree. These steps correspond to the microfill and consistency operations in Figure 2.

Let $R_{k,j}$ be the j-th maximal run with one valid repaired identity $q_{k,j} \geq 0$. Its sampled-frame extent is

$$R_{k,j} = [s_{k,j}, e_{k,j}] = \text{MaxRun}_j(\tilde{\mathbf{m}}_k) \tag{6}$$

where $s_{k,j}$ and $e_{k,j}$ are the first and last sampled-frame indices in the run. Let $\tau_{k,i}$ be the absolute timestamp of sampled frame i. The clip begins at $\tau_{k,s_{k,j}}$ and ends at the timestamp of the next sampled frame. A run reaching the last sampled frame ends at t_k^c. Denoting these boundaries by $\tau_{k,j}^s$ and $\tau_{k,j}^e$ gives

$$P_{k,j} = \{I_t \mid \tau_{k,j}^s \leq t < \tau_{k,j}^e\} \tag{7}$$

Thus, the repaired identity sequence determines a maximal run, the run determines temporal boundaries, and the boundaries determine the passenger clip. Adjacent run boundaries within three sampled frames are merged, while clips shorter than 0.5 seconds are removed. The retained clips become the inputs to passenger direction filtering.

3.3 Complex-to-Simple Passenger Direction Mapping

We assume that only boarding passengers proceed to fare payment, so non-boarding passengers are filtered out. Direct activity recognition, however, requires transit-specific spatiotemporal knowledge that a general-purpose VLM may lack. We therefore introduce a Complex-to-Simple (CS) mapping that reformulates the task as a generic facing-direction judgment.

Each candidate clip $P_{k,j}$ contains one temporally isolated passenger. We uniformly sample $M = 4$ chronological frames to form $\mathcal{F}k, j$ and submit only this sequence to the VLM. Given the orientation-focused prompt q_{face}, the VLM produces

$$\hat{o}_{k,j} = \Phi_{\text{face}}(\mathcal{F}_{k,j}, q_{\text{face}}) \tag{8}$$

where Φ_{face} denotes the VLM for this task, and $\hat{o}_{k,j}$ is its visual-state judgment from $O = \{\text{front}, \text{side}, \text{back}, \text{inside}\}$. A fixed rule ψ_{CS} then maps this generic judgment to passenger direction

$$\hat{a}_{k,j} = \psi_{\text{CS}}(\hat{o}_{k,j}) = \begin{cases} \text{boarding}, & \hat{o}_{k,j} \in \{\text{front}, \text{side}\}, \\ \text{alighting}, & \hat{o}_{k,j} = \text{back}, \\ \text{staying}, & \hat{o}_{k,j} = \text{inside} \end{cases} \tag{9}$$

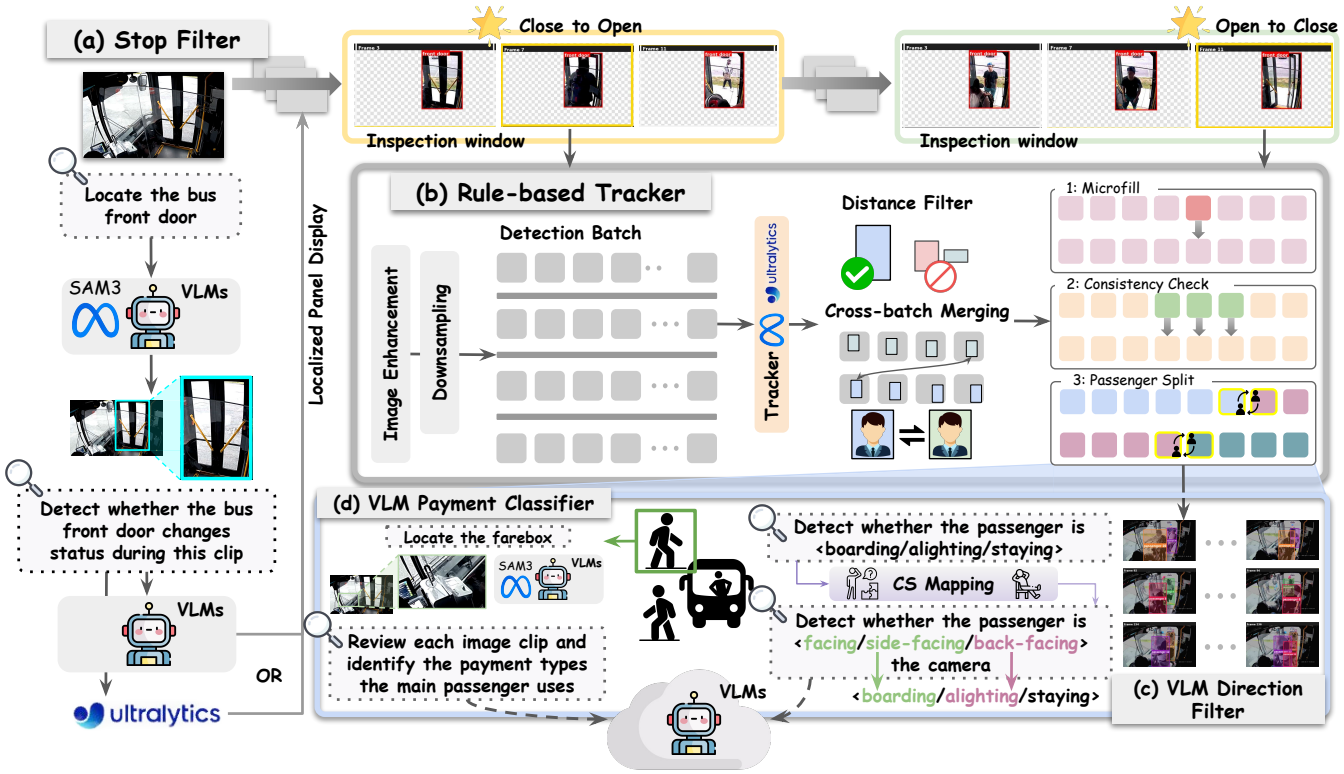

Figure 2: Overview of GHR-VLM. The edge pipeline grounds the front-door region and restricts downstream processing to stop intervals during which the door is open. A rule-based tracker then generates passenger-specific clips, a VLM-based direction filter retains only boarding passengers, and a farebox-grounded two-stage VLM assigns payment labels.

This rule injects the fixed inward-facing camera geometry as domain knowledge, so the VLM only judges visual orientation. The boarding gate is

$$g_{k,j} = \mathbb{I}[\hat{a}_{k,j} = \text{boarding}] \tag{10}$$

where $\mathbb{I}$ is the indicator function. Only clips with $g_{k,j} = 1$ proceed to payment classification.

3.4 Spatio-Temporal Grounded Payment Classification

The retained clip provides passenger-level temporal grounding, but payment evidence may appear only briefly within a small farebox region. We therefore introduce a two-stage coarse-to-fine procedure for payment spatiotemporal grounding.

Shared farebox grounding. Let P_{k_0,j_0} be the first retained passenger clip and $P_{k_0,j_0}[0]$ its first frame. SAM 3 localizes the farebox using the concept prompt q_f = "farebox payment box":

$$B_f = \Psi_{\text{SAM3}}(P_{k_0,j_0}[0], q_f) \tag{11}$$

where $B_f = (x_1, y_1, x_2, y_2)$ contains normalized corner coordinates and is cached for subsequent clips. Stage-specific margins produce $B_f^{(1)}$ and $B_f^{(2)}$. Stage 1 uses $(0.35, 0.35, -0.50)$ for broader context, whereas Stage 2 uses $(0, 0.10, -0.50)$ for tighter localization. The negative bottom adjustment removes the lower portion of the detected box.

Stage 1 coarse classification and evidence selection. We uniformly sample 16 frames over the complete clip, crop each with $B_f^{(1)}$, and arrange them chronologically in a 4×4 contact sheet $G_{k,j}^{(1)}$. The first VLM call returns

$$(\tilde{y}_{k,j}, \tilde{c}_{k,j}, \tilde{\rho}_{k,j}, \mathcal{E}_{k,j}) = \Phi_{\text{pay}}(G_{k,j}^{(1)}, q_{\text{pay}}) \tag{12}$$

where Φ_{pay} is the VLM and q_{pay} defines the five labels in $\mathcal{Y}$ together with their visual rules. The outputs $\tilde{y}_{k,j}$, $\tilde{c}_{k,j}$, and $\tilde{\rho}_{k,j}$ are the provisional label, confidence, and rationale. The key design here is to ask the VLM to select the evidence frame set $\mathcal{E}_{k,j} \subseteq 1, \ldots, 16$ that supports its reasoning, which reveals the inner cues of how the VLM makes its judgment.

Stage 2 grounded refinement and final classification. For clearer inspection of the key evidence, the earliest and latest evidence frames define a refined temporal interval for further reasoning. If Stage 1 returns no valid evidence frame, the full clip is used. We then uniformly resample 16 frames from the selected interval, crop them with $B_f^{(2)}$, and construct $G_{k,j}^{(2)}$. The second VLM call applies the same visual rules to the refined evidence

$$(\hat{y}_{k,j}, \hat{c}_{k,j}, \hat{\rho}_{k,j}) = \Phi_{\text{pay}}(G_{k,j}^{(2)}, q_{\text{pay}}) \tag{13}$$

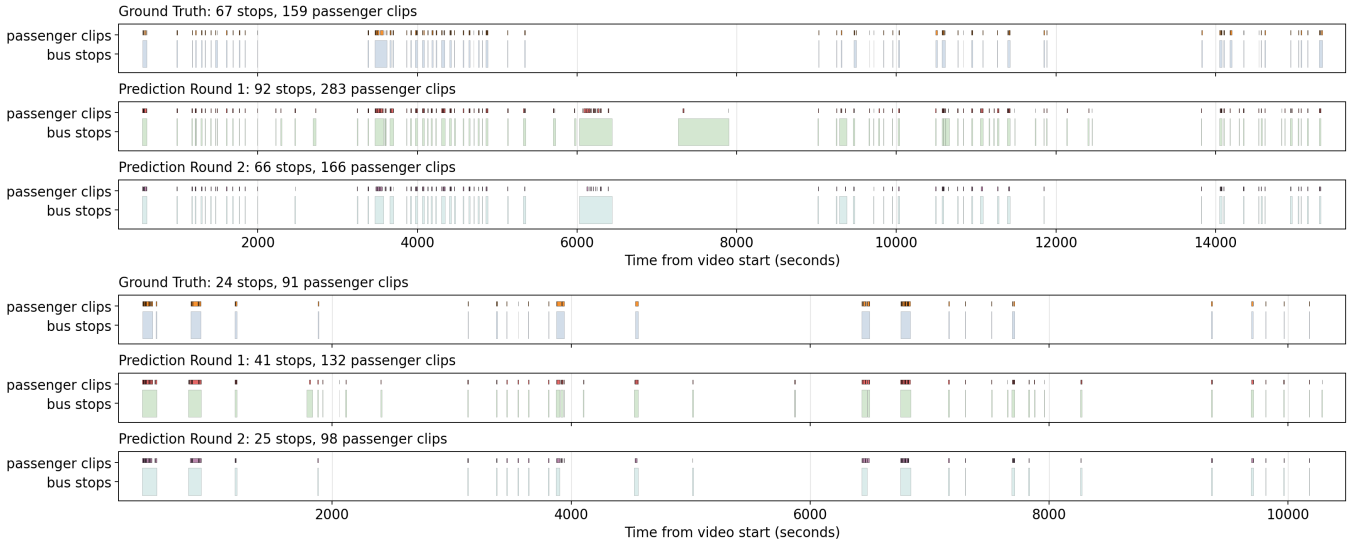

Figure 3: Full-video timelines of detected stops and passenger clips for C3_1 (top) and C3_3 (bottom). Each panel compares the ground truth with the raw Round 1 predictions and the refined Round 2 predictions after direction filtering and stop merging.

Table 1: Stop- and passenger-clip interval detection using one-to-one matching at temporal IoU ≥ 0.2. The best precision (Prec.), recall (Rec.), and F1 score for each video are shown in bold.

	Predicted Count		Stop Intervals			Passenger Clips		
Round	**Stops**	**Clips**	**Prec. ↑**	**Rec. ↑**	**F1 ↑**	**Prec. ↑**	**Rec. ↑**	**F1 ↑**
C3_1	*GT: 67 stops, 159 passenger clips*							
R1	92	283	0.663	**0.910**	0.767	0.505	**0.899**	0.647
R2	66	166	**0.894**	0.881	**0.887**	**0.687**	0.717	**0.702**
C3_3	*GT: 24 stops, 91 passenger clips*							
R1	41	132	0.561	**0.958**	0.708	0.636	**0.923**	0.753
R2	25	98	**0.880**	0.917	**0.898**	**0.816**	0.879	**0.847**

where $\hat{y}_{k,j} \in \mathcal{Y}$, $\hat{c}_{k,j}$, and $\hat{\rho}_{k,j}$ denote the final payment label, confidence, and rationale. Stage 1 searches the full passenger clip, whereas Stage 2 concentrates temporal sampling and spatial resolution on the selected evidence. The final decision is thus grounded in both the passenger clip and the shared farebox geometry, without requiring payment-specific model training.

4 Experiments

4.1 Dataset and Implementation Details

We collect 436 minutes of real onboard bus surveillance video, including two video records: C3_1 was recorded during daytime operation from 8 a.m. to 2 p.m., while C3_3 was recorded from 6 p.m. to 9 p.m. with nighttime scenes accounting for nearly half of the footage. Both videos have a resolution of 1280×720 at 10 FPS. We manually annotate every stop interval and passenger clip with its payment label. In our edge-cloud implementation, lightweight edge models monitor the door, track passengers, and segment clips, while cloud VLMs process only the grounded passenger clips and farebox contact sheets. We use GPT-4o for behavior classification and GPT-5.4-mini for payment classification. All experiments run on one NVIDIA RTX 4090 GPU with 24 GB of memory.

4.2 Evaluation of Edge Modules

We evaluate stop monitoring, passenger tracking, and passenger segmentation over the complete timelines. Predictions are matched one to one with ground truth at temporal IoU ≥ 0.2. At this stage, we obtain results from two rounds. Round 1 includes all detected stops and rule-based passenger clips. Round 2 removes non-boarding clips and merges neighboring stops when their retained clips are separated by less than 15 seconds. As shown in Figure 3 and Table 1, Round 1 achieves high recall but overpredicts both stops and passenger clips because it retains non-boarding passengers and fragmented intervals. Round 2 brings the predicted counts close to the ground truth and removes many isolated intervals. Stop F1 increases from 0.767 to 0.887 on C3_1 and from 0.708 to 0.898 on C3_3, while passenger-clip F1 rises from 0.647 to 0.702 and from 0.753 to 0.847.

This precision gain comes at the cost of lower stop-level and passenger-level recall. The primary bottleneck is the wrong removal of true boarding, resulting in an inherent precision-recall

Table 2: Stage-wise payment classification results on C3_1 and C3_3. Prec. and Rec. denote class-wise precision and recall, respectively. Arrows in Stage 2 indicate changes relative to Stage 1.

	Overall	Cash		QR		Swipe		Tap		Evade	
Video / Stage	**Acc.**	**Prec.**	**Rec.**	**Prec.**	**Rec.**	**Prec.**	**Rec.**	**Prec.**	**Rec.**	**Prec.**	**Rec.**
C3_1 / Stage 1	0.349	0.400	0.455	0.185	0.556	0.062	0.091	0.400	0.196	0.867	0.491
C3_1 / Stage 2	0.313↓	0.480↑	0.545↑	0.187↑	0.778↑	0.000↓	0.000↓	0.280↓	0.137↓	0.905↑	0.358↓
C3_3 / Stage 1	0.485	0.400	0.308	0.324	0.750	0.333	0.056	0.407	0.688	0.950	0.559
C3_3 / Stage 2	0.536↑	0.571↑	0.308	0.381↑	0.500↓	0.400↑	0.333↑	0.429↑	0.750↑	0.846↓	0.647↑

Binary Evade/Non-Evade Classification					
Video / Stage	**Acc.**	**Evade Prec.**	**Evade Rec.**	**Non-Evade Prec.**	**Non-Evade Rec.**
C3_1 / Stage 1	0.813	0.867	0.491	0.801	0.965
C3_1 / Stage 2	0.783↓	0.905↑	0.358↓	0.766↓	0.982↑
C3_3 / Stage 1	0.837	0.950	0.559	0.808	0.984
C3_3 / Stage 2	0.837	0.846↓	0.647↑	0.833↑	0.938↓

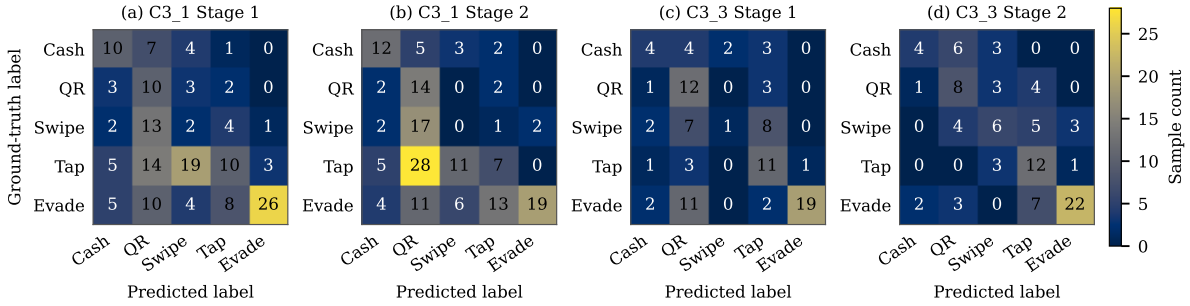

Figure 4: Payment-type confusion matrices for both VLM stages on C3_1 and C3_3. Each cell reports the number of samples from its ground-truth row assigned to the predicted column.

trade-off. Manual inspection further attributes the remaining errors to atypical activities, including the driver leaving the cockpit (at around 6,000 seconds in C3_1), simultaneous boarding events, and passengers reappearing in the scene, etc. Direct sunlight in C3_1 also causes image blur and a darkened bus interior, further degrading tracking accuracy.

4.3 Evaluation of VLM Modules

The VLM-based payment classification results are presented in Table 2 and Figure 4. Overall, the system performs better on C3_3 than on C3_1, consistent with the edge-module results. C3_1 exhibits more challenging illumination, with direct sunlight and shadows reducing the visibility of passengers and farebox interactions. In contrast, C3_3 has more consistent lighting, enabling the VLM to recognize payment behaviors more reliably.

On C3_3, the two-stage VLM improves five-class accuracy from 0.485 to 0.536, with higher recall for Swipe, Tap, and Evade. In contrast, Stage 2 reduces five-class accuracy on C3_1 from 0.349 to 0.313. For binary evasion detection, C3_1 accuracy decreases from 0.813 to 0.783, while Evade recall drops from 0.491 to 0.358. On C3_3, accuracy remains unchanged at 0.837, but Evade recall increases from 0.559 to 0.647. These results indicate that, although Stage 2 provides more focused evidence, its effectiveness still depends strongly on visual quality.

The confusion matrices further reveal different refinement effects across the two videos. On C3_1, Stage 2 concentrates more errors in the QR column, whereas on C3_3 it reduces confusion involving QR and Tap. This difference explains why refinement benefits the visually more consistent C3_3 sequence but not C3_1. The VLM also struggles to distinguish QR from Tap because of their visual similarity, and Swipe is particularly difficult to recognize due to the small swipe region and subtle motion. Nevertheless, refinement increases Swipe recall on C3_3 from nearly zero to 0.333, suggesting that more focused evidence can help the VLM capture fine-grained payment actions.

5 Conclusions

We propose GHR-VLM, a grounded hybrid framework that combines lightweight model-based perception with selective VLM reasoning for zero-shot transit video analytics. Model-based modules extract structured spatiotemporal passenger evidence, enabling VLMs to classify payment behavior without payment-specific training. The resulting pipeline transforms a single onboard surveillance stream into stop-, passenger-, and payment-level events. Experiments on real-world bus videos demonstrate the value of explicit grounding, although fine-grained payment recognition remains challenging. Future work will strengthen VLM reasoning and robustness under degraded video conditions.

References

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. 2021. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*. 6836–6846.

[2] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and J Qwen-VL Zhou. 2023. A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966* 6 (2023), 3.

[3] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. 2025. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631* (2025).

[4] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. 2021. Is space-time attention all you need for video understanding?. In *Icml*, Vol. 2. 4.

[5] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. 2025. Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719* (2025).

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European conference on computer vision*. Springer, 213–229.

[7] José Correa, Tobias Harks, Vincent JC Kreuzen, and Jannik Matuschke. 2017. Fare evasion in transit networks. *Operations research* 65, 1 (2017), 165–183.

[8] Amir Dib, Noëlie Cherrier, Martin Graive, Baptiste Rérolle, and Eglantine Schmitt. 2023. Unified occupancy on a public transport network through combination of AFC and APC data. In *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 1963–1970.

[9] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. 2019. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*. 6202–6211.

[10] Kaicong Huang, Talha Azfar, Jack Reilly, and Ruimin Ke. 2025. Transitreid: Transit od data collection with occlusion-resistant dynamic passenger reidentification. *arXiv preprint arXiv:2504.11500* (2025).

[11] Kaicong Huang, Weiheng Oh, Thomas Guggisberg, and Ruimin Ke. 2026. iPay: Integrated Payment Action Recognition via Multimodal Networks and Adaptive Spatial Prior Learning. *arXiv preprint arXiv:2605.10732* (2026).

[12] Nico Jahn and Michael Siebert. 2022. Engineering the neural automatic passenger counter. *Engineering Applications of Artificial Intelligence* 114 (2022), 105148.

[13] Glenn Jocher, Jing Qiu, and Ayush Chaurasia. 2024. Ultralytics yolo11.

[14] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*. 4015–4026.

[15] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*. PMLR, 19730–19742.

[16] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. 2022. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 10965–10975.

[17] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. 2024. Video-llava: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 conference on empirical methods in natural language processing*. 5971–5984.

[18] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems* 36 (2023), 34892–34916.

[19] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. 2024. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*. Springer, 38–55.

[20] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. 2024. Video-chatgpt: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 12585–12602.

[21] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. 2022. Simple open-vocabulary object detection. In *European conference on computer vision*. Springer, 728–755.

[22] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. 2025. Sam 2: Segment anything in images and videos. In *International Conference on Learning Representations*, Vol. 2025. 28085–28128.

[23] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. 2024. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159* (2024).

[24] Wei-Qing Ren, Yu-Ben Qu, Chao Dong, Yu-Qian Jing, Hao Sun, Qi-Hui Wu, and Song Guo. 2023. A survey on collaborative DNN inference for edge intelligence. *Machine Intelligence Research* 20, 3 (2023), 370–395.

[25] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, et al. 2024. Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18221–18232.

[26] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. 2022. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems* 35 (2022), 10078–10093.

[27] Johannes van der Vyver. 2024. A Deep Neural Network Approach to Fare Evasion. *arXiv preprint arXiv:2405.17855* (2024).

[28] Mengmeng Wang, Jiazheng Xing, and Yong Liu. 2021. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472* (2021).

[29] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. 2017. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*. IEEE, 3645–3649.

[30] Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

[31] Hang Zhang, Xin Li, and Lidong Bing. 2023. Video-llama: An instruction-tuned audio-visual language model for video understanding. In *Proceedings of the 2023 conference on empirical methods in natural language processing: system demonstrations*. 543–553.

[32] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. 2022. Bytetrack: Multi-object tracking by associating every detection box. In *European conference on computer vision*. Springer, 1–21.