ISOCOMM: HETEROGENEOUS COOPERATIVE PERCEPTION ACROSS TRAINING-ISOLATED SYSTEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Heterogeneous cooperative perception enables agents with different sensing modalities and perception models to collaborate by sharing complementary information. Recent methods have moved heterogeneous cooperation closer to real-world deployment by relaxing assumptions on modalities, architectures, and system extensibility. However, cross-agent compatibility is still commonly established before deployment through shared source training domains, coupled model components, or pre-established representations and communication protocols. We study a more open setting where complete perception systems are developed independently on private domains and future collaborators are unknown. We formulate this setting as **Training-Isolated Heterogeneous Cooperative Perception**. Under this setting, conventional Direct fusion cannot reliably turn an unseen sender into a cooperative gain, because the sender representation and receiver fusion interface have been learned through separate development processes. To address this challenge, we propose **IsoComm** with two mechanisms: Protocol-Basis Diversification (PBD) and Ownership-Preserving Attention (OPA). PBD diversifies the communication bases encountered during local training and reduces dependence on a specific local basis. OPA preserves sender-native value encoding while keeping relevance estimation and aggregation under receiver control. This design enables zero-shot cooperation between independently trained systems without pair-specific optimization. For persistent partnerships, we further introduce Communication-Aware Commissioning to calibrate lightweight communication interfaces while keeping both perception systems frozen. Across three unseen senders, IsoComm consistently improves over the ego-only receiver, achieving 88.32–88.54 AP_{50} versus 82.49 for ego-only, while reducing the cross-sender AP_{50} range to 0.22 points and supporting annotation-efficient commissioning.

1 INTRODUCTION

Cooperative perception enables connected autonomous agents to overcome the physical limitations of single-agent sensing by sharing information across complementary viewpoints. Such collaboration is particularly valuable under occlusion, long-range sparsity, and restricted fields of view, where information missing from one agent may remain directly observable to another (Yu et al., 2022; Xu et al., 2023c). Among different communication paradigms, intermediate feature fusion offers an attractive compromise between the information richness of raw-data sharing and the communication efficiency of decision-level fusion. Real-world agents, however, are unlikely to share identical perception systems. Vehicles and infrastructure developed by different manufacturers may employ different sensing modalities, model architectures, objectives, and training pipelines. This has motivated growing interest in heterogeneous cooperative perception (HCP), which seeks to make heterogeneous intermediate representations interoperable.

Recent HCP research has progressively relaxed the assumptions required for heterogeneous agents to collaborate, bringing cooperative perception closer to realistic deployment. Earlier approaches (Xu et al., 2023a;b; Xiang et al., 2023) typically jointly optimize a predefined collection of heterogeneous agents, whereas extensible frameworks (Lu et al., 2024; Gao et al., 2025; Liu et al., 2026; Si et al., 2025; Yang et al., 2026; Luo et al., 2024; Xia et al., 2025) such as HEAL (Lu et al., 2024) allow newly introduced agents to align with an established collaborative feature space with only limited retraining cost. STAMP (Gao et al., 2025) further broadens agent heterogeneity beyond

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

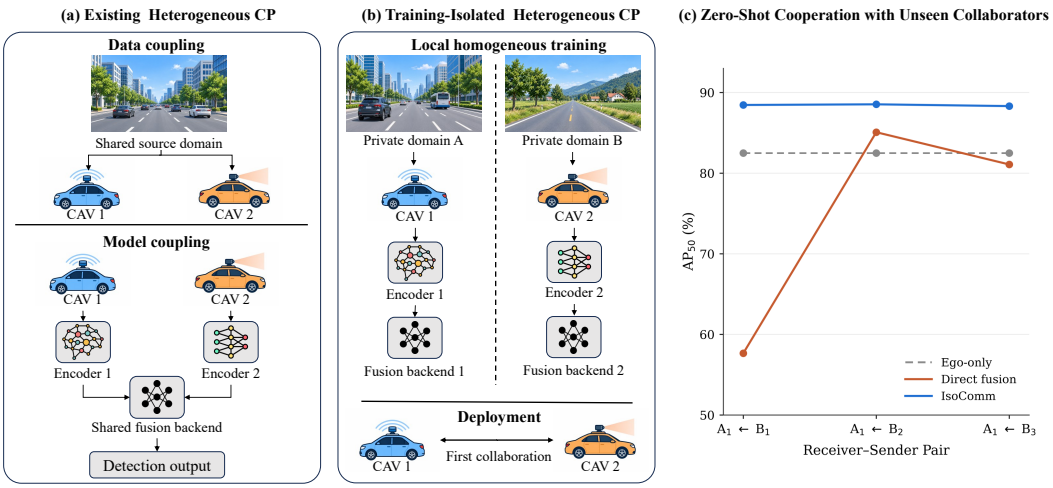

Figure 1: Training-isolated heterogeneous cooperative perception. (a) Common HCP development patterns establish compatibility through shared training context or coupled model components and representation interfaces. (b) In the training-isolated setting, complete perception systems are developed independently in private domains and first encounter one another at deployment. (c) With receiver A_1 fixed, Direct fusion exhibits strongly pair-dependent performance across unseen collaborators B_1–B_3, whereas IsoComm maintains stable AP_{50}; dashed lines denote ego-only performance.

sensing modalities and model architectures to differences in perception tasks, while CodeAlign (Liu et al., 2026) relaxes modality co-occurrence requirements through modality-specific code spaces and cross-modal translation. Another line of work moves compatibility learning toward deployment: PHCP (Si et al., 2025) adapts lightweight feature interfaces from unlabeled collaborator observations, and BOLT (Yang et al., 2026) enables cooperation between independently trained detectors through online feature adaptation. Together, these advances substantially reduce the coordination required for heterogeneous collaboration. Yet, as illustrated in Fig. 1(a), existing HCP systems typically acquire a degree of cross-agent compatibility before deployment because heterogeneous agents are developed either within a shared source training domain or around coupled model components. Such compatibility is therefore partly engineered by the development process itself. This dependence limits direct cooperation between independently developed systems when they first encounter one another, as cross-vendor deployments may provide neither shared training context nor coupled model components in advance.

To characterize first-encounter cooperation between independently developed systems, we formulate a new problem setting termed **Training-Isolated Heterogeneous Cooperative Perception**. As illustrated in Fig. 1(b), different vendors develop complete perception systems independently on private source domains. They share neither source training data nor jointly optimized cross-vendor perception or fusion components, and no common latent communication protocol is established in advance. Future collaborators are also unknown during development. At deployment, the underlying perception systems remain frozen. We consider two regimes: zero-shot cooperation, where previously unseen systems collaborate at their first encounter without pair-specific adaptation data, target labels, or deployment-time optimization; and persistent-partnership commissioning, where a lightweight communication interface may subsequently be calibrated while both base perception systems remain frozen.

Under this setting, the central challenge is that Direct fusion cannot reliably turn an unseen sender into a cooperative gain. Here, Direct fusion follows the conventional receiver-side fusion paradigm: the sender transmits its intermediate feature, which is directly processed by the receiver's own fusion network. As shown in Fig. 1(c), we fix receiver A_1, trained on OPV2V (Xu et al., 2022c), and vary the sender among B_1–B_3, which are independently trained on a separate dataset, V2XREAL (Xiang et al., 2024). The ego-only receiver achieves 82.49 AP_{50}, while Direct fusion ranges from 57.66 with B_1 to 85.08 with B_2, crossing the ego-only baseline. The same receiver can therefore be either helped or harmed depending on the encountered system pair. This makes robustness to unknown pair compatibility a central design objective for training-isolated cooperation, motivating

a communication interface that can accommodate unseen collaborators without relying on favorable pairings.

The observations above suggest two requirements for zero-shot cooperation with unknown future collaborators. First, the receiver's fusion rule should not become overly specialized to the representation basis encountered during its own local training, since an unseen sender may communicate through a different basis. Second, when a new sender is introduced, its transmitted content should preserve the value encoding learned by that sender rather than being re-encoded entirely through the receiver's local interface. Based on these principles, we introduce **IsoComm**, which prepares the communication interface locally for unknown future collaborators. During isolated local training, Protocol-Basis Diversification (PBD) exposes the fusion module to diverse protocol bases, reducing its dependence on a specific locally learned representation basis. Ownership-Preserving Attention (OPA) keeps relevance estimation and downstream aggregation under receiver control while preserving sender-native value encoding. At zero-shot deployment, independently trained systems are composed directly without pair-specific optimization, and PBD is disabled at inference. Across the three unseen senders in Fig. 1(c), IsoComm achieves 88.32–88.54 AP_{50}, consistently improving over the 82.49 ego-only receiver and reducing the cross-sender range to only 0.22 points. For persistent partnerships, Communication-Aware Commissioning (CAC) further calibrates lightweight communication interfaces while keeping both base perception systems frozen. With 40 selected labeled frames, CAC reaches 90.76 ± 0.16 AP_{50}, surpassing random selection with 60 frames at 90.44 ± 0.28 while using 33.3% fewer labeled frames. Our contributions are summarized as follows:

- We formulate Training-Isolated Heterogeneous Cooperative Perception, which studies cooperation between complete perception systems independently developed on private training domains without shared source data, jointly optimized cross-vendor components, or a pre-established communication protocol.
- We introduce IsoComm, a unified framework that prepares communication interfaces during isolated local training and enables zero-shot cooperation with unseen collaborators through Protocol-Basis Diversification and Ownership-Preserving Attention, with optional Communication-Aware Commissioning for persistent partnerships.
- Extensive experiments on the OPV2V and V2XREAL datasets demonstrate the effectiveness of IsoComm in enabling robust zero-shot cooperation across unseen heterogeneous collaborators and further improving persistent partnerships through annotation-efficient commissioning.

2 Related Work

2.1 Cooperative Perception

Cooperative perception enables connected agents to overcome occlusion, limited sensing range, and sparse observations by exchanging complementary information (Wang et al., 2020; Li et al., 2021; Hu et al., 2023). Existing methods can be broadly categorized into early, late, and intermediate fusion according to the stage at which information is shared. Early fusion exchanges raw sensor observations at high communication cost, while late fusion communicates only perception outputs and discards much of the intermediate semantic information. Intermediate-feature fusion provides a better balance between perception quality and communication efficiency and has become the dominant paradigm (Chen et al., 2019; Wang et al., 2020; Li et al., 2021; Xu et al., 2022a;b). F-Cooper (Chen et al., 2019) is an early feature-based cooperative perception system, while V2VNet (Wang et al., 2020) introduces learned message passing for vehicle-to-vehicle collaboration. Coopernaut extends feature communication to end-to-end cooperative driving with compact point-based messages (Cui et al., 2022). Who2com (Liu et al., 2020b) and When2com (Liu et al., 2020a) further study selective communication among agents, and Where2comm (Hu et al., 2022) extends this idea to spatially informative regions. Related work also studies communication efficiency and latency through compact or selective feature exchange (Lei et al., 2022; Hu et al., 2024; Yang et al., 2023). Other studies improve robustness to pose errors (Vadivelu et al., 2021; Lu et al., 2023) and communication asynchrony (Wei et al., 2023). These methods establish effective collaborative fusion mechanisms, but

typically assume compatible feature interfaces across agents, an assumption that becomes restrictive when perception systems are developed independently.

2.2 Heterogeneous Cooperative Perception

Heterogeneous cooperative perception relaxes the assumption that all agents share identical sensing modalities or model architectures. HM-ViT (Xiang et al., 2023) introduces heterogeneous attention for cross-modal feature fusion, while MPDA (Xu et al., 2023b) reduces representation gaps through pair-specific feature adaptation. BM2CP studies LiDAR–camera collaboration through modality-guided fusion, whereas DI-V2X addresses sensor-induced domain gaps through domain-invariant distillation and adaptive fusion (Zhao et al., 2023; Li et al., 2024). Recent work further studies how heterogeneous agents can be incorporated into a collaborative system with lower integration cost. HEAL (Lu et al., 2024) establishes a unified collaborative feature space and aligns new agents to it through backward alignment, while STAMP (Gao et al., 2025) maps agent-specific representations through a shared protocol domain. NegoCollab (Shao et al., 2026) learns a negotiated common representation from heterogeneous agent features instead of choosing one agent's representation as the reference, and GT-Space (Wang et al., 2026) constructs a common feature space from ground-truth labels and projects heterogeneous features into this shared reference. GenComm (Zhou et al., 2025) takes a generative approach, using conditional feature generation to reconstruct sender information in the receiver's semantic space, while CodeAlign (Liu et al., 2026) connects isolated modalities through modality-specific code spaces and cross-modal translation. Another line of work shifts compatibility learning to deployment: PHCP (Si et al., 2025) adapts lightweight feature interfaces from unlabeled collaborator observations, and BOLT (Yang et al., 2026) performs online adaptation between independently trained detectors. These advances reduce the coordination required for heterogeneous collaboration, but compatibility is still typically established either before deployment through shared or coupled representations, or after encounter through pair-specific adaptation. Training-Isolated Heterogeneous Cooperative Perception instead studies first-encounter cooperation between independently developed systems without a pre-established cross-agent interface or pair-specific optimization.

3 Problem Formulation

3.1 Training-Isolated Heterogeneous Cooperative Perception

Consider a set of agents $\mathcal{A}$ that perform intermediate-feature cooperative perception. The perception system of agent i is

$$\mathcal{M}_i = (E_i, F_i, H_i), \tag{1}$$

where E_i, F_i, and H_i denote its sensor encoder, cooperative fusion module, and perception head, respectively. These systems may use different sensing modalities, architectures, fusion designs, and perception objectives.

Given an observation x_i, agent i extracts a feature $z_i = E_i(x_i)$. A sender j transmits its feature to receiver r, where it is spatially aligned by

$$z_{j\to r} = \Gamma_{j\to r}(z_j). \tag{2}$$

For a sender set $\mathcal{N}_r$, the receiver predicts

$$\hat{y}_r = H_r\left(F_r\left(z_r, \{z_{j\to r}\}_{j\in\mathcal{N}_r}\right)\right). \tag{3}$$

Here, $\Gamma_{j\to r}$ represents message transmission and geometric alignment. Feature fusion and final prediction are performed by the receiver.

Each system is trained on a private source-domain dataset $\mathcal{D}_i$:

$$\theta_i^* = \arg\min_{\theta_i} \mathbb{E}_{(x,y)\sim\mathcal{D}_i}\left[\mathcal{L}_i(\theta_i; x, y)\right]. \tag{4}$$

Training isolation imposes three conditions. First, $\mathcal{D}_i$ remains private to its owner. Other vendors cannot access it for joint training or compatibility learning. Second, each system has its own encoder, fusion module, perception head, and learned communication representation. The systems share no jointly optimized perception components, common latent space, or cross-vendor feature translator.

Third, the identity of a future collaborator is unknown during training. The system therefore receives no pair-specific alignment, interface training, or fusion tuning for that collaborator.

These conditions jointly characterize training-isolated heterogeneous cooperative perception. Heterogeneous systems developed with shared source data, joint representation alignment, shared cooperative components, or knowledge of their future partners fall outside this setting.

At deployment, receiver r may encounter a previously unseen sender s:

$$z_r = E_r(x_r), \qquad z_{s\to r} = \Gamma_{s\to r}\left(E_s(x_s)\right). \tag{5}$$

The representation produced by E_s and the fusion rule implemented by F_r originate from separate training processes. Their cross-agent compatibility is therefore uncontrolled by construction. A shared semantic basis between z_r and $z_{s\to r}$ cannot be assumed at deployment.

The base perception systems remain frozen after deployment. In particular, the encoders, fusion modules, and perception heads are not jointly retrained across vendors. Permissible deployment-time interaction depends on the cooperation regime defined below.

3.2 Deployment Regimes

Ad-hoc zero-shot cooperation. Receiver r encounters a previously unseen sender s and begins cooperation at the first encounter. The model pair uses no pair-specific training data, target-domain labels, deployment-time optimization, or cross-agent calibration. All perception and fusion components remain frozen.

Persistent-partnership commissioning. Two independently trained systems form a persistent deployment partnership and may perform a one-time commissioning stage using newly collected deployment frames and a small manually labeled calibration set. Commissioning updates only the lightweight communication interface; the private source datasets $\mathcal{D}_r$ and $\mathcal{D}_s$ remain inaccessible, and all base perception components remain frozen.

4 Methodology

4.1 IsoComm Overview

Following Section 3, agent i encodes its observation x_i into a communication feature $z_i = E_i(x_i)$. For each neighbor $j \in \mathcal{N}_i$, where $\mathcal{N}_i$ denotes the collaborator set of i, $\Gamma_{j\to i}$ aligns z_j to the coordinate frame of i. The fusion module F_i aggregates the aligned features into h_i, and the detection head H_i predicts $\hat{y}_i$.

IsoComm prepares the interface during isolated source-domain training. PBD perturbs non-ego communication features before fusion,

$$x_j \xrightarrow{E_j} z_j \xrightarrow{\text{PBD}} \bar{z}_j \xrightarrow{\Gamma_{j\to i}} z_{j\to i} \xrightarrow{F_i} h_i \xrightarrow{H_i} \hat{y}_i, \tag{6}$$

where $\bar{z}_j$ is the perturbed feature. Each system also trains native query, key, and value projections under the OPA-compatible architecture. With local collaborators, OPA reduces to native attention because each neighbor uses the same system's W_V. Fig. 2 illustrates zero-shot collaborator substitution and frozen-checkpoint composition without an architectural change. PBD is disabled at inference. For persistent partnerships, CAC trains residual key and value adapters on selected newly labeled deployment frames while the base systems remain frozen.

4.2 Zero-Shot Cooperation across Isolated Representations

Training isolation leaves compatibility between independently learned communication features uncontrolled and pair-dependent. PBD reduces specialization to a single local basis, while OPA retains each sender's native value interface.

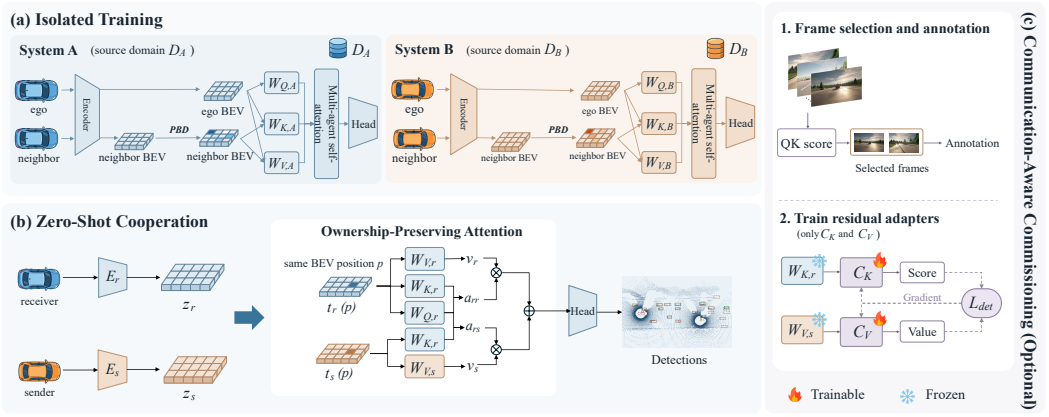

Figure 2: IsoComm lifecycle. (a) Each private-domain system establishes PBD and OPA-compatible attention during local homogeneous training. (b) Zero-shot deployment substitutes a sender encoder and native value projection while the receiver retains query, key, relation, aggregation, and detection components; PBD is off. (c) Optional CAC selects frames using frozen QK activations, obtains labels, and trains only residual C_K and C_V adapters.

Protocol-Basis Diversification. Let $z_j \in \mathbb{R}^{C\times H\times W}$ be a non-ego communication feature and $z_j^\flat \in \mathbb{R}^{C\times HW}$ its spatially flattened form. PBD samples

$$G_j \sim \mathcal{N}(0,1)^{C\times C}, \quad S_j = \frac{G_j - G_j^\top}{2}, \quad Q_j = \exp(\alpha S_j), \tag{7}$$

where G_j is Gaussian, S_j is skew-symmetric, α controls the perturbation magnitude, and Q_j is therefore an orthogonal channel-basis rotation. Channel-wise scale and offset are sampled as

$$s_{jc} = \max(0.1, 1+\beta u_{jc}), \qquad b_{jc} = \beta\epsilon_{jc}, \quad u_{jc}\sim\mathcal{U}(-1,1), \quad \epsilon_{jc}\sim\mathcal{N}(0,1), \tag{8}$$

where c indexes channels and β controls the affine perturbation. The transformed feature is

$$\mathcal{T}_j(z_j) = \text{reshape}\left[Q_j\left(\text{diag}(s_j)z_j^\flat + b_j\mathbf{1}^\top\right)\right], \tag{9}$$

where $s_j, b_j \in \mathbb{R}^C$ and $\mathbf{1}\in\mathbb{R}^{HW}$. We use $\bar{z}_i = z_i$ for the ego feature and $\bar{z}_j = \mathcal{T}_j(z_j)$ for $j\in\mathcal{N}_i$. Applying the same transformation at every spatial location retains BEV layout and pose geometry, without guaranteeing semantic preservation.

Let $\mathcal{D}_i$ be the private training set of system i, θ_i its model parameters, and $\mathcal{L}_i$ its detection loss. Training follows

$$\min_{\theta_i} \mathbb{E}_{(x,y)\sim\mathcal{D}_i,\mathcal{T}}\left[\mathcal{L}_i\left(H_i\left(F_i\left(\{\Gamma_{j\to i}(\bar{z}_j)\}_{j\in\{i\}\cup\mathcal{N}_i}\right)\right), y\right)\right]. \tag{10}$$

PBD introduces no trainable parameters or auxiliary loss, and each system is trained only on its own source domain.

Ownership-Preserving Attention. After geometric alignment, cross-agent attention operates on tokens at corresponding BEV positions. Let t_i be the ego token at one position and t_j the aligned token from collaborator j at the corresponding position. For head $m\in\{1,\dots,M\}$, with head dimension d_m, the local system computes

$$q_i^m = W_{Q,i}^m t_i, \qquad k_j^m = W_{K,i}^m t_j, \qquad v_j^m = W_{V,i}^m t_j, \tag{11}$$

where $W_{Q,i}^m$, $W_{K,i}^m$, and $W_{V,i}^m$ are its query, key, and value projections. The receiver-owned relation matrices $R_{\text{att},i}^m$ and $R_{\text{msg},i}^m$ respectively modify attention scoring and value messages:

$$a_{ij}^m = \text{softmax}_j\left(\frac{(q_i^m)^\top R_{\text{att},i}^m k_j^m}{\sqrt{d_m}}\right), \qquad \mu_{ij}^m = R_{\text{msg},i}^m v_j^m. \tag{12}$$

The weighted messages are combined by the output projection $W_{O,i}$. During local training, all collaborators share the same locally trained model, so the neighbor value path naturally uses $W_{V,i}^m$.

At deployment, let r be the receiver and s an independently trained sender. Direct fusion applies the receiver value projection to the sender token t_s,

$$v_s^{m,\text{direct}} = W_{V,r}^m t_s. \tag{13}$$

OPA instead preserves the sender-native value projection:

$$v_j^{m,\text{OPA}} = \begin{cases} W_{V,r}^m t_r, & j = r, \\ W_{V,s(j)}^m t_j, & j \neq r, \end{cases} \tag{14}$$

where $s(j)$ denotes the system providing neighbor j. The receiver still computes $q_r^m = W_{Q,r}^m t_r$ and $k_j^m = W_{K,r}^m t_j$, and retains $R_{\text{att},r}^m$, $R_{\text{msg},r}^m$, $W_{O,r}$, the remaining fusion layers, and H_r. Thus, the receiver determines relevance and downstream aggregation, while the sender retains its native value encoding.

Zero-shot IsoComm requires no new deployment-time architecture or optimization. It replaces a local collaborator with the unseen sender while retaining the locally established OPA ownership rule. We additionally limit the aggregated neighbor-message norm relative to the ego message as a conservative zero-shot safeguard; this bound is disabled after commissioning.

4.3 Communication-Aware Commissioning

For a persistent receiver–sender pair, optional CAC uses selected newly labeled deployment frames to calibrate residual communication mismatch without retraining the perception systems. CAC is not required for zero-shot cooperation.

Communication-aware acquisition. For deployment frame f, let Ω_f contain BEV positions where both receiver and pose-aligned sender feature energies exceed their respective 75th percentiles. This shared-activation proxy does not use a ground-truth visibility mask. Using M attention heads, CAC computes

$$c_{\text{QK}}(f) = \frac{1}{M|\Omega_f|} \sum_{m=1}^{M} \sum_{p \in \Omega_f} \cos\left(q_r^{m,f}(p), k_s^{m,f}(p)\right), \qquad d_{\text{QK}}(f) = 1 - c_{\text{QK}}(f), \tag{15}$$

where p indexes spatial positions, the superscript f identifies frame-specific tokens, and d_{QK} measures receiver-query and receiver-projected sender-key mismatch. The score itself uses only frozen communication activations and poses, without detection supervision. Frames are ranked within each scenario, then selected with scenario balancing and temporal separation before annotation.

Lightweight interface calibration. Let C_K and C_V be residual key and value adapters with parameters θ_{C_K} and θ_{C_V}. For the receiver-projected sender key $K_s = W_{K,r} t_s$ and sender-native value $V_s = W_{V,s} t_s$, CAC computes

$$K_s' = K_s + \Delta_K(K_s; \theta_{C_K}), \qquad V_s' = V_s + \Delta_V(V_s; \theta_{C_V}), \tag{16}$$

where Δ_K is a pointwise full-channel bottleneck and Δ_V is a spatial full-channel bottleneck. Both adapters are residual and identity-initialized.

Let $\mathcal{D}_{r,s}^{\text{com}}$ denote the labeled commissioning set and y_r the receiver-side detection target. CAC optimizes

$$\min_{\theta_{C_K}, \theta_{C_V}} \mathbb{E}_{(x_r, x_s, y_r) \sim \mathcal{D}_{r,s}^{\text{com}}} \left[\mathcal{L}_{\text{det}} \left(H_r \left(F_r^{\text{OPA}} \left(z_r, z_{s \to r}; C_K, C_V \right) \right), y_r \right) \right], \tag{17}$$

where F_r^{OPA} denotes the receiver fusion module using the OPA value interface. Only θ_{C_K} and θ_{C_V} are updated; all encoders, attention projections, relation matrices, remaining fusion layers, and the detection head remain frozen. The resulting interface is shared by deployment instances of the same model pair.

Table 1: Cross-domain cooperation with A_1, an OPV2V-trained PointPillar receiver with a deep backbone, and three independently trained V2XREAL senders. B_1 uses PointPillar with a shallow backbone, while B_2 and B_3 use SECOND and PIXOR, respectively. Direct fusion and IsoComm are the only methods that strictly satisfy our zero-shot protocol, which uses no pair-specific adaptation data, deployment-time optimization, or target labels. Pair Adapt. indicates adaptation using unlabeled observations from the encountered pair.

Method	Pair Adapt.	B_1			B_2			B_3		
		AP_{30}	AP_{50}	AP_{70}	AP_{30}	AP_{50}	AP_{70}	AP_{30}	AP_{50}	AP_{70}
Ego-only	–	84.42	82.49	70.06	84.42	82.49	70.05	84.42	82.49	70.05
Direct fusion	✗	57.84	57.66	53.41	86.89	85.08	72.49	81.83	81.08	69.56
MPDA (Xu et al., 2023b)	✓	76.63	76.29	67.16	89.94	87.96	74.05	87.95	86.60	72.62
CodeFilling (Hu et al., 2024)	✓	62.95	62.72	57.53	86.90	85.04	72.35	80.11	79.57	69.54
BOLT (Yang et al., 2026)	✓	85.44	82.29	67.92	86.65	83.19	68.81	86.35	82.95	68.63
IsoComm	✗	90.68	88.46	74.22	90.66	88.54	74.36	90.30	88.32	74.31

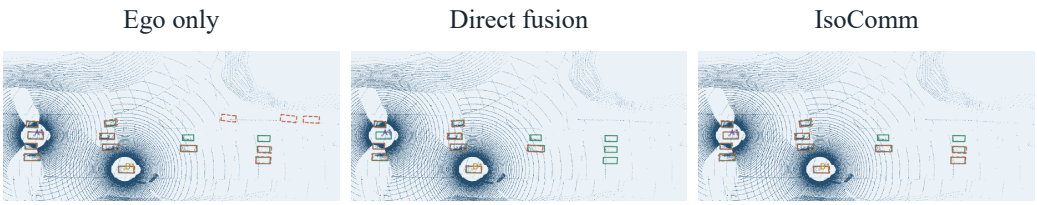

Figure 3: Qualitative comparison of zero-shot cooperation with an unseen collaborator. IsoComm preserves ego-side detections while recovering complementary targets that are missed by Direct fusion. Ground-truth boxes are shown in green solid lines and predictions in red dashed lines.

5 Experiments

5.1 Experimental Settings

We use OPV2V (Xu et al., 2022c) and V2XREAL (Xiang et al., 2024) as private training domains A and B. A_i and B_j denote complete perception systems independently trained in the two domains, with detailed dataset and system configurations provided in Appendix A.1. Our main evaluation fixes A_1, an OPV2V PointPillar receiver with a deep BEV backbone, and pairs it with three independently trained V2XREAL senders: B_1, a PointPillar system with a shallow BEV backbone, B_2 based on SECOND, and B_3 based on PIXOR. The pairs $A_1 \leftarrow B_1$, $A_1 \leftarrow B_2$, and $A_1 \leftarrow B_3$ keep the receiver, fusion backend, detection head, and evaluation protocol fixed, and are evaluated on the OPV2V test split with an x/y half-range of $102.4/51.2$ m. Each system is trained only on its own private source domain. PBD and the OPA-compatible attention interface are established during local training, and PBD is disabled at inference. Zero-shot deployment composes frozen receiver and sender checkpoints without pair-specific adaptation data, parameter optimization, or target labels. The communicated BEV feature has 128 channels and a spatial resolution of 64×128. We report AP_{30}, AP_{50}, and AP_{70} at IoU thresholds 0.3, 0.5, and 0.7.

We first compare against Ego-only perception and Direct fusion, which respectively measure single-agent performance and the effect of directly connecting independently trained communication interfaces. IsoComm and Direct fusion follow our strict zero-shot protocol. To broaden comparison with existing heterogeneous collaboration methods under a common setting, we additionally introduce a controlled relaxation after isolated training: the evaluated receiver–sender pair may use unlabeled pair observations for adaptation, while target ground-truth labels remain unavailable. Under this protocol, we compare with MPDA (Xu et al., 2023b), CodeFilling (Hu et al., 2024), and BOLT (Yang et al., 2026), all of which can optimize without target annotations. We use the domain-adversarial objective of MPDA, the codebook objective of CodeFilling, and the ego-detection objective of BOLT for pair-specific adaptation. For persistent partnerships, CAC starts from zero-shot IsoComm and optimizes only the residual key and value adapters while keeping the encoders, backbones, fusion network, sender value projections, and detection head frozen. Commissioning results are averaged over three seeds; the adapters are trained for 10 epochs with Adam using an initial learning rate of 10^{-3}, reduced by a factor of 0.1 after epoch 5.

Table 2: Communication-aware commissioning on the persistent $A_1 \leftarrow B_1$ pair. Limited-budget results are averaged over three seeds. Random uses scenario-diverse random sampling, while CAC uses scenario-normalized QK mismatch with scenario balancing and temporal separation.

Selection	Frames	Labeled boxes	AP_{30}	AP_{50}	AP_{70}
Zero-shot	0	0	90.68	88.46	74.22
Random	40	633.3 ± 4.6	93.30 ± 0.55	90.19 ± 0.38	73.89 ± 0.64
CAC	40	622	94.25 ± 0.27	90.76 ± 0.16	74.09 ± 0.19
Random	60	915.7 ± 28.0	93.57 ± 0.30	90.44 ± 0.28	73.98 ± 0.28
CAC	60	935	94.53 ± 0.11	91.01 ± 0.07	74.46 ± 0.22
Full commissioning	Full	–	96.09	93.01	76.75

5.2 Cross-Domain Cooperation with Unseen Senders

Table 1 evaluates cross-domain cooperation with receiver A_1 fixed and the unseen sender varied among B_1, B_2, and B_3. Direct fusion ranges from 57.66 to 85.08 AP_{50} and crosses the 82.49 Ego-only baseline, showing that an unseen sender can either improve or degrade the same receiver. IsoComm instead improves over Ego-only for all three senders, achieving 88.32–88.54 AP_{50} with a cross-sender range of only 0.22 points. On $A_1 \leftarrow B_1$, it improves AP_{50} by 30.80 points over Direct fusion. Figure 3 shows that IsoComm preserves reliable ego-side detections while recovering complementary targets. Compared with MPDA, CodeFilling, and BOLT, which may use unlabeled pair observations for adaptation, IsoComm achieves the highest AP_{50} and AP_{70} across all three senders without pair-specific adaptation.

5.3 Communication-Aware Commissioning

We evaluate optional persistent-partnership commissioning on $A_1 \leftarrow B_1$, the pair with the largest Direct-fusion degradation relative to Ego-only. Table 2 compares the 40- and 60-frame budgets. CAC-40 reaches 90.76 ± 0.16 AP_{50} with 40 labeled frames and 622 boxes, exceeding Random-60 at 90.44 ± 0.28 with 60 frames and 915.7 ± 28.0 boxes. This is 33.3% fewer labeled frames and approximately 32.1% fewer annotated boxes. Full commissioning reaches 93.01 AP_{50} without retraining either base system.

5.4 Ablation Study

Table 3 evaluates the individual contributions of PBD and OPA on $A_1 \leftarrow B_1$. Direct fusion reaches 57.66 AP_{50}. PBD alone raises this result to 86.25 by reducing specialization to the local communication basis, while OPA alone reaches 80.24 by retaining the sender-native value projection. Combining the two components reaches 88.46 AP_{50} and 74.22 AP_{70}. Relative to PBD alone, OPA contributes another 2.98 points at AP_{30} and 2.21 points at AP_{50}. These results show that PBD and OPA address distinct aspects of the zero-shot interface.

Table 3: Ablation of PBD and OPA on $A_1 \leftarrow B_1$.

Method	AP_{30}	AP_{50}	AP_{70}
Direct fusion	57.84	57.66	53.41
PBD	87.70	86.25	73.64
OPA	81.92	80.24	60.29
PBD + OPA	90.68	88.46	74.22

6 Conclusion

We introduced Training-Isolated Heterogeneous Cooperative Perception, a setting in which complete perception systems are developed independently on private domains, with no shared source data, jointly optimized cross-vendor components, or pre-established communication protocol. Under this setting, conventional Direct fusion cannot reliably turn an unseen sender into a cooperative gain. To address this challenge, we proposed IsoComm, which prepares PBD and OPA-compatible communication interfaces during isolated local development and enables zero-shot cooperation through frozen-checkpoint composition. For persistent partnerships, Communication-Aware Commissioning further calibrates lightweight key and value interfaces while keeping the base perception systems frozen. Experiments across unseen heterogeneous collaborators show consistent cooperative gains in zero-shot deployment and annotation-efficient calibration for persistent partnerships.

AI USE STATEMENT

Generative AI tools assisted literature retrieval; implementation and debugging of selected training, evaluation, and analysis code; organization and checking of recorded experimental results; and manuscript polishing, and translation. All quantitative results reported in the paper come from recorded model training and evaluation outputs rather than generated data. AI-assisted content was reviewed against the implementation, experiment logs, and cited sources. The authors retain responsibility for the scientific claims, software, experimental results, and final manuscript.

REFERENCES

Qi Chen, Xu Ma, Sihai Tang, Jingda Guo, Qing Yang, and Song Fu. F-cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3d point clouds. In *Proceedings of the ACM/IEEE Symposium on Edge Computing*, pp. 88–100, 2019.

Jiaxun Cui, Hang Qiu, Dian Chen, Peter Stone, and Yuke Zhu. Coopernaut: End-to-end driving with cooperative perception for networked vehicles. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 17252–17262, 2022.

Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on Robot Learning (CoRL)*, pp. 1–16, 2017.

Xiangbo Gao, Runsheng Xu, Jiachen Li, Ziran Wang, Zhiwen Fan, and Zhengzhong Tu. Stamp: Scalable task-and model-agnostic collaborative perception. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, volume 2025, pp. 54656–54676, 2025.

Yue Hu, Shaoheng Fang, Zixing Lei, Yiqi Zhong, and Siheng Chen. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. In *Advances in Neural Inf. Process. Syst.(NIPS)*, volume 35, pp. 4874–4886, 2022.

Yue Hu, Yifan Lu, Runsheng Xu, Weidi Xie, Siheng Chen, and Yanfeng Wang. Collaboration helps camera overtake lidar in 3d detection. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 9243–9252, 2023.

Yue Hu, Juntong Peng, Sifei Liu, Junhao Ge, Si Liu, and Siheng Chen. Communication-efficient collaborative perception via information filling with codebook. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 15481–15490, 2024.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014.

Zixing Lei, Shunli Ren, Yue Hu, Wenjun Zhang, and Siheng Chen. Latency-aware collaborative perception. In *Proc. Eur. Conf. Comp. Vis. (ECCV)*, pp. 316–332, 2022.

Xiang Li, Junbo Yin, Wei Li, Chengzhong Xu, Ruigang Yang, and Jianbing Shen. DI-V2X: Learning domain-invariant representation for vehicle-infrastructure collaborative 3d object detection. In *Proc. Conf. AAAI.*, volume 38, pp. 3208–3215, 2024. doi: 10.1609/aaai.v38i4.28105.

Yiming Li, Shunli Ren, Pengxiang Wu, Siheng Chen, Chen Feng, and Wenjun Zhang. Learning distilled collaboration graph for multi-agent perception. In *Advances in Neural Inf. Process. Syst.(NIPS)*, volume 34, pp. 29541–29552, 2021.

Changxing Liu, Zichen Chao, and Siheng Chen. Linking modality isolation in heterogeneous collaborative perception. *arXiv:2603.00609*, 2026.

Yen-Cheng Liu, Junjiao Tian, Nathaniel Glaser, and Zsolt Kira. When2com: Multi-agent perception via communication graph grouping. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 4105–4114, 2020a.

Yen-Cheng Liu, Junjiao Tian, Chih-Yao Ma, Nathan Glaser, Chia-Wen Kuo, and Zsolt Kira. Who2com: Collaborative perception via learnable handshake communication. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 6876–6883, 2020b.

Yifan Lu, Quanhao Li, Baoan Liu, Mehrdad Dianati, Chen Feng, Siheng Chen, and Yanfeng Wang. Robust collaborative 3d object detection in presence of pose errors. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 4812–4818, 2023.

Yifan Lu, Yue Hu, Yiqi Zhong, Dequan Wang, Yanfeng Wang, and Siheng Chen. An extensible framework for open heterogeneous collaborative perception. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, volume 2024, pp. 44576–44593, 2024.

Tianyou Luo, Quan Yuan, Guiyang Luo, Yuchen Xia, Yujia Yang, and Jinglin Li. Plug and play: A representation enhanced domain adapter for collaborative perception. In *Proc. Eur. Conf. Comp. Vis. (ECCV)*, pp. 287–303, 2024.

Congzhang Shao, Quan Yuan, Guiyang Luo, Yue Hu, Danni Wang, Liu Yilin, Rui Pan, Bo Chen, and Jinglin Li. Negocollab: A common representation negotiation approach for heterogeneous collaborative perception. In *Advances in Neural Inf. Process. Syst.(NIPS)*, volume 38, pp. 87196–87217, 2026.

Hao Si, Ehsan Javanmardi, and Manabu Tsukada. You share beliefs, i adapt: Progressive heterogeneous collaborative perception. In *Proc. IEEE Int. Conf. Comp. Vis. (ICCV)*, pp. 27521–27530, 2025.

Nicholas Vadivelu, Mengye Ren, James Tu, Jingkang Wang, and Raquel Urtasun. Learning to communicate and correct pose errors. In *Conference on Robot Learning (CoRL)*, pp. 1195–1210, 2021.

Tsun-Hsuan Wang, Sivabalan Manivasagam, Ming Liang, Bin Yang, Wenyuan Zeng, and Raquel Urtasun. V2vnet: Vehicle-to-vehicle communication for joint perception and prediction. In *Proc. Eur. Conf. Comp. Vis. (ECCV)*, pp. 605–621, 2020.

Wentao Wang, Haoran Xu, and Guang Tan. Gt-space: Enhancing heterogeneous collaborative perception with ground truth feature space. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2026.

Sizhe Wei, Yuxi Wei, Yue Hu, Yifan Lu, Yiqi Zhong, Siheng Chen, and Ya Zhang. Asynchrony-robust collaborative perception via bird's eye view flow. In *Advances in Neural Inf. Process. Syst.(NIPS)*, volume 36, pp. 28462–28477, 2023.

Yuchen Xia, Quan Yuan, Guiyang Luo, Xiaoyuan Fu, Yang Li, Xuanhan Zhu, Tianyou Luo, Siheng Chen, and Jinglin Li. One is plenty: A polymorphic feature interpreter for immutable heterogeneous collaborative perception. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 1592–1601, 2025.

Hao Xiang, Runsheng Xu, and Jiaqi Ma. Hm-vit: Hetero-modal vehicle-to-vehicle cooperative perception with vision transformer. In *Proc. IEEE Int. Conf. Comp. Vis. (ICCV)*, pp. 284–295, 2023.

Hao Xiang, Zhaoliang Zheng, Xin Xia, Runsheng Xu, Letian Gao, Zewei Zhou, Xu Han, Xinkai Ji, Mingxi Li, Zonglin Meng, et al. V2x-real: a large-scale dataset for vehicle-to-everything cooperative perception. In *Proc. Eur. Conf. Comp. Vis. (ECCV)*, pp. 455–470, 2024.

Runsheng Xu, Yi Guo, Xu Han, Xin Xia, Hao Xiang, and Jiaqi Ma. Opencda: an open cooperative driving automation framework integrated with co-simulation. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pp. 1155–1162. IEEE, 2021.

Runsheng Xu, Zhengzhong Tu, Hao Xiang, Wei Shao, Bolei Zhou, and Jiaqi Ma. Cobevt: Cooperative bird's eye view semantic segmentation with sparse transformers. In *Conference on Robot Learning (CoRL)*, 2022a.

Runsheng Xu, Hao Xiang, Zhengzhong Tu, Xin Xia, Ming-Hsuan Yang, and Jiaqi Ma. V2x-vit: Vehicle-to-everything cooperative perception with vision transformer. In *Proc. Eur. Conf. Comp. Vis. (ECCV)*, pp. 107–124, 2022b.

Runsheng Xu, Hao Xiang, Xin Xia, Xu Han, Jinlong Li, and Jiaqi Ma. Opv2v: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 2583–2589, 2022c.

Runsheng Xu, Weizhe Chen, Hao Xiang, Xin Xia, Lantao Liu, and Jiaqi Ma. Model-agnostic multi-agent perception framework. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 1471–1478, 2023a.

Runsheng Xu, Jinlong Li, Xiaoyu Dong, Hongkai Yu, and Jiaqi Ma. Bridging the domain gap for multi-agent perception. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2023b.

Runsheng Xu, Xin Xia, Jinlong Li, Hanzhao Li, Shuo Zhang, Zhengzhong Tu, Zonglin Meng, Hao Xiang, Xiaoyu Dong, Rui Song, Hongkai Yu, Bolei Zhou, and Jiaqi Ma. V2V4Real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 13712–13722, 2023c.

Dingkang Yang, Kun Yang, Yuzheng Wang, Jing Liu, Zhi Xu, Rongbin Yin, Peng Zhai, and Lihua Zhang. How2comm: Communication-efficient and collaboration-pragmatic multi-agent perception. In *Advances in Neural Inf. Process. Syst.(NIPS)*, volume 36, 2023. doi: 10.52202/075280-1093.

Kang Yang, Tianci Bu, Peng Wang, Deying Li, and Yongcai Wang. Bolt: Online lightweight adaptation for preparation-free heterogeneous cooperative perception. *arXiv:2605.00405*, 2026.

Haibao Yu, Yizhen Luo, Mao Shu, Yiyi Huo, Zebang Yang, Yifeng Shi, Zhenglong Guo, Hanyu Li, Xing Hu, Jirui Yuan, and Zaiqing Nie. DAIR-V2X: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 21361–21370, 2022.

Binyu Zhao, Wei Zhang, and Zhaonian Zou. BM2CP: Efficient collaborative perception with LiDAR-camera modalities. In *Conference on Robot Learning (CoRL)*, volume 229, pp. 1022–1035, 2023.

Junfei Zhou, Penglin Dai, Quanmin Wei, Bingyi Liu, Xiao Wu, and Jianping Wang. Pragmatic heterogeneous collaborative perception via generative communication mechanism. In *Advances in Neural Inf. Process. Syst.(NIPS)*, volume 38, pp. 54484–54509, 2025.

A Appendix

This appendix provides the dataset, system, training, commissioning, and evaluation details that support the additional experiments and reproducibility of IsoComm.

A.1 Datasets and System Configurations

A.1.1 Source domains and data access

OPV2V. OPV2V (Xu et al., 2022c) provides simulated cooperative driving observations collected with CARLA (Dosovitskiy et al., 2017) and OpenCDA (Xu et al., 2021). Its multimodal setting includes LiDAR and camera observations; the dataset description in CodeAlign (Liu et al., 2026) reports one LiDAR and four monocular cameras per connected vehicle. The loader exposes 6,374 training, 1,980 validation, and 2,170 test frames from 44, 9, and 16 physical scenarios, respectively. At most five connected autonomous vehicles (CAVs) are used in one frame. The experiments detect the vehicle class only. Systems A_1–A_4 use the OPV2V training and validation splits for isolated local development, and formal AP evaluation uses the test split.

For the $A_1 \leftarrow B_1$ commissioning experiments, we reuse OPV2V training frames as an experimental proxy for the paired observations that would become available after a persistent partnership is established. The base systems are first developed independently and frozen before commissioning; CAC then uses paired multi-agent observations only for post-development interface calibration and does not update either base perception system. The OPV2V validation split is used for commissioning checkpoint selection, and final AP is evaluated on the disjoint test split.

For the reported limited-budget experiments, the commissioning candidate pool is restricted to training frames that contain at least one annotated target, include sender B_1, and yield a valid QK mismatch score. This produces 5,659 candidate frames from 42 scenarios. The ground-truth requirement is used only to define the experimental calibration pool; the QK acquisition score itself is computed from frozen communication features and poses without using detection labels. Full commissioning uses all 6,374 OPV2V training frames as the calibration proxy.

V2XREAL. V2XREAL (Xiang et al., 2024) is a real-world cooperative perception dataset containing vehicle and infrastructure observations. With `dataset_mode=vc`, the loader-visible training, validation, and test splits contain 47/5,816, 6/717, and 8/996 scenarios/frames, respectively. Agent identifiers $-1, -2, 1, 2$ cover vehicle and infrastructure agents, and at most five agents are used per frame.

The V2XREAL sender systems are trained with three superclasses. *Vehicle* maps `LongVehicle`, `Car`, and `PoliceCar`; *pedestrian* maps `Child`, `RoadWorker`, `Pedestrian`, `Scooter`, `ScooterRider`, `Motorcycle`, `MotorcycleRider`, and `BicycleRider`; and *truck* maps `Truck`, `Van`, `TrashCan`, `ConcreteTruck`, and `Bus`. These categories supervise isolated V2XREAL development. Cross-domain evaluation always uses the receiver-owned OPV2V detection head.

A.1.2 Complete system inventory

Table 4 records all seven systems in the inventory. A_2 is included for completeness but has no additional appendix result. A_1 and B_1 belong to the PointPillar family, but their BEV backbone depths and training domains differ. Their common encoder family therefore does not identify them as the same architecture or checkpoint. The camera systems use Lift-Splat-Shoot (LSS) with the image backbones named in Table 4.

Shared communication and fusion interface. Every system exposes a 128-channel communication feature. Our OPA is implemented based on V2XViT (Xu et al., 2022b). V2X-ViT contains three transformer layers with one fusion block per layer. Agent attention uses eight heads of dimension 32, giving 256-dimensional internal query, key, and value projections. The spatial pyramid uses head counts $[16, 8, 4]$, head dimensions $[16, 32, 64]$, and window sizes $[4, 8, 16]$. Receiver-owned detection uses separate 1×1 convolutional classification, regression, and direction heads.

Table 4: Perception-system inventory. Each row denotes a complete independently trained system with its own fusion and detection components.

System	Source domain	Sensor	Encoder / image backbone
A_1	OPV2V	LiDAR	PointPillar (Deep BEV backbone)
A_2	OPV2V	Camera	LSS–EfficientNet
A_3	OPV2V	LiDAR	SECOND
A_4	OPV2V	Camera	LSS–ResNet101
B_1	V2XREAL	LiDAR	PointPillar (Shallow BEV backbone)
B_2	V2XREAL	LiDAR	SECOND
B_3	V2XREAL	LiDAR	PIXOR

A-domain systems. A_1 uses PointPillar over $x \in [-102.4, 102.4]$, $y \in [-51.2, 51.2]$, and $z \in [-3, 1]$ m. Pillars are $0.4 \times 0.4 \times 4$ m, contain at most 32 points, and are limited to 32,000/70,000 pillars in training/test. The PFN and scatter output 64 channels. Its three BEV stages use block counts $[3, 5, 8]$, channels $[64, 128, 256]$, and strides $[2, 2, 2]$; three 128-channel deconvolution branches form a 384-channel BEV tensor. A 3×3, stride-2 shrinker produces the final $128 \times 64 \times 128$ communication feature.

A_2 uses LSS with an ImageNet-pretrained EfficientNet-B0 over `camera0–camera3`. Images are resized from 600×800 to 384×512; training uses resize range $[0.65, 0.70]$, rotation range $[-3.6°, 3.6°]$, bottom crop $[0, 0.05]$, and no horizontal flip. Lifted depth uses 48 linearly increasing discretization bins over $[2, 50]$ m. The LSS BEV grid covers $x, y \in [-51.2, 51.2]$ m at 0.4 m resolution and uses the same three-stage BEV-backbone family and shrinker as A_1, producing $128 \times 64 \times 64$. A_2 is included only in the inventory.

A_3 uses SECOND over the same range as A_1, with $0.1 \times 0.1 \times 0.1$ m voxels, at most five points per voxel, and 32,000/70,000 training/test voxels. MeanVFE receives four point features. VoxelBackBone8x uses sparse channel widths 16/32/64/64 and outputs 64 sparse channels; HeightCompression produces 128 channels, followed by an identity BEV backbone and a 3×3, stride-2 shrinker. The communication feature is $128 \times 64 \times 128$.

A_4 uses LSS with a non-pretrained ResNet-101 and the same four camera views. Images are resized from 600×800 to 336×448; training resize lies in $[0.56, 0.61]$, with the same rotation, bottom-crop, and no-flip policy family as A_2. It uses 48 depth bins over $[2, 50]$ m and a 0.4 m BEV grid over $x, y \in [-51.2, 51.2]$ m, producing $128 \times 64 \times 64$. For $A_4 \leftarrow B_1$, the sender encoder is instantiated on the same square range, so both communication maps are 64×64; no learned grid converter is used.

B-domain systems. B_1 is a PointPillar system trained on V2XREAL over $x \in [-102.4, 102.4]$, $y \in [-51.2, 51.2]$, and $z \in [-15, 15]$ m. It uses $0.4 \times 0.4 \times 30$ m pillars, at most 32 points per pillar, and 64,000/640,000 training/test pillars. Its shallow BEV backbone has one three-block, stride-2, 64-channel stage and a 128-channel upsampling branch. A 3×3, stride-2 shrinker produces $128 \times 64 \times 128$. When paired with A_1, deployment preprocessing adopts the receiver-compatible z-range $[-3, 1]$ m and 4 m pillar height while retaining the representation weights learned independently on V2XREAL. This geometric instantiation does not use cross-system cooperative training.

B_2 uses SECOND over $x \in [-102.4, 102.4]$, $y \in [-51.2, 51.2]$, and $z \in [-3, 1]$ m, with $0.1 \times 0.1 \times 0.1$ m voxels, five points per voxel, and 32,000/70,000 training/test voxels. Its MeanVFE, VoxelBackBone8x, HeightCompression, and 3×3, stride-2 shrinker follow the structural family of A_3, producing $128 \times 64 \times 128$.

B_3 uses a dense PIXOR BEV over the same range as B_2 at 0.4 m resolution. Its input has shape $512 \times 256 \times 11$. The backbone uses block counts $[3, 6, 6, 3]$, an initial 32 channels, stage widths 24/48/64/96, and a 96-channel top-down output. An independently trained 1×1 projection maps 96 channels to the final $128 \times 64 \times 128$ communication tensor.

Table 5: OPA-compatible isolated base-training schedules. Each system uses only its own source-domain training and validation data.

System	A_1	A_2	A_3	A_4	B_1	B_2	B_3
Epochs	30	30	30	30	40	40	40
Batch size	2	1	2	1	2	1	1

A.2 Implementation Details

A.2.1 Isolated local training

Local development consists of OPA-compatible base training followed by a PBD continuation, both restricted to the system's own source domain. The base schedules are shown in Table 5. All systems use Adam (Kingma & Ba, 2014) with initial learning rate 2×10^{-3}, $(\beta_1, \beta_2) = (0.9, 0.999)$, $\epsilon = 10^{-10}$, and weight decay 10^{-4}. MultiStepLR applies a factor of 0.1 at epochs 10 and 15. The checkpoint with the lowest validation loss initializes a 20-epoch PBD continuation with batch size one and the same optimizer and scheduler family.

For A-domain systems, focal classification uses $\alpha = 0.25$, $\gamma = 2$, and weight 2; SmoothL1 regression uses $\sigma = 3$ and weight 2; and direction cross-entropy has weight 0.2. Camera systems additionally use depth-loss weight 1. B-domain training uses three-class classification weight 1 and regression weight 2. The main point-cloud augmentation for A_1 and B_1–B_3 comprises a world x-flip, rotation in $[-\pi/4, \pi/4]$, and scale in $[0.95, 1.05]$. The A_3 configuration has no data augmentation; camera augmentation follows the image settings above.

A.2.2 PBD implementation

PBD is applied after the modality encoder, backbone, and shrinker, and before pose alignment and fusion. On every forward pass, each non-ego agent independently samples one transform; the ego feature is never transformed. For a $C \times H \times W$ communication feature,

$$G \sim \mathcal{N}(0,1)^{C\times C}, \qquad S = \frac{\alpha}{2}(G - G^\top), \qquad Q = \mathrm{matrix_exp}(S),$$

with $\alpha = 0.5$. Per-channel scale and offset are sampled as

$$s = \mathrm{clamp}\left(1 + \beta(2U - 1),\, \mathrm{min} = 0.1\right), \qquad b = \beta\xi, \tag{18}$$

where

$$U \sim \mathcal{U}(0,1)^C, \qquad \xi \sim \mathcal{N}(0,1)^C, \qquad \beta = 0.3.$$

The vectors $s, b \in \mathbb{R}^C$ are broadcast as $[C, 1, 1]$. A single sampled Q, s, and b is shared across all spatial positions of the corresponding feature map. The implementation uses a contiguous $C \times (HW)$ flattening for the channel-space transformation and then restores the original $C \times H \times W$ layout. PBD is active only during local training and is disabled in evaluation mode, including zero-shot deployment, acquisition, and CAC.

A.2.3 OPA implementation

Before the outer warp, the 128-channel communication BEV is concatenated with a three-channel all-zero prior. Inside the V2X-ViT encoder, the prior is separated again and attention operates on the 128 communication channels. Each transformer layer applies $\mathrm{Linear}(128, 256)$ query, key, and value projections and reshapes the result into eight heads of dimension 32. Thus, the 256-dimensional CAC interface is the native attention-projection space rather than a separate learned converter.

In every transformer layer, the receiver supplies W_Q, W_K, its ego W_V, R_{att}, R_{msg}, W_O, and the downstream fusion and detection path. An unseen sender key is still produced by receiver W_K, whereas its value uses the sender-native W_V. The receiver output projection is $\mathrm{Linear}(256, 128)$. At zero-shot composition, the sender provides its encoder/backbone/shrinker trunk, and its isolated fusion checkpoint is loaded separately only to supply the native per-layer W_V modules for the non-ego path. The receiver provides the shared fusion and prediction path. No pair-specific optimization is performed. Identity STTF and the zero prior are fixed execution paths rather than learned components.

A.2.4 CONSERVATIVE ZERO-SHOT SAFEGUARD

The zero-shot safeguard is applied independently in every attention block, for the ego query, at each spatial position and attention head. The ego and neighbor messages are 32-dimensional head vectors after relation-message projection and attention aggregation. Define

$$u_{\text{ego}} = a_{rr}\mu_{rr}, \qquad u_{\text{nei}} = \sum_{j \neq r} a_{rj}\mu_{rj}, \tag{19}$$

$$\lambda = \min\left(1, \frac{\rho\|u_{\text{ego}}\|_2}{\max(\|u_{\text{nei}}\|_2, 10^{-6})}\right), \qquad o_r = u_{\text{ego}} + \lambda u_{\text{nei}}, \qquad \rho = 0.05. \tag{20}$$

The L2 norm is taken over the final 32-dimensional feature axis. Safeguarded heads are concatenated only afterward and passed through receiver W_O. The safeguard is enabled for zero-shot IsoComm, is absent from Direct fusion, and is disabled for CAC training and commissioned inference.

A.3 COMMUNICATION-AWARE COMMISSIONING

A.3.1 ADAPTER ARCHITECTURE

The key adapter C_K is

$$\text{Linear}(256, 64) \to \text{LayerNorm}(64) \to \text{GELU} \to \text{Linear}(64, 256),$$

with bias in both linear layers and residual output

$$K' = K + \Delta_K(K).$$

The weight and bias of the final linear layer are zero-initialized, so the residual branch is initialized to zero. This adapter contains 33,216 trainable parameters.

For the spatial value adapter C_V, let N denote the number of agent entries retained along the attention-agent dimension. The eight attention heads are merged from

$$[8, H, W, N, 32]$$

to

$$[N, 256, H, W],$$

so that the spatial adapter operates on the complete 256-dimensional value representation of each agent. Its residual branch is

$$\begin{aligned}&\text{Conv}_{1\times1}(256, 64) \to \text{GN}_8 \to \text{GELU} \to \text{Conv}_{3\times3}(64, 64)\\ &\quad\to \text{GN}_8 \to \text{GELU} \to \text{Conv}_{3\times3}(64, 64) \to \text{GN}_8 \to \text{GELU} \to \text{Conv}_{1\times1}(64, 256).\end{aligned} \tag{21}$$

Both 3×3 convolutions use padding one, and all convolutions use bias. Only the weight and bias of the final 1×1 up-projection are zero-initialized. The adapted value is obtained by residual addition,

$$V' = V + \Delta_V(V).$$

The spatial value adapter contains 107,328 trainable parameters.

The final CAC interface jointly trains C_K and the spatial full-channel C_V, for a total of

$$33{,}216 + 107{,}328 = 140{,}544$$

trainable parameters. All encoders, backbones, shrinkers, receiver-owned fusion and detection components, and the sender-native value projections remain frozen.

A.3.2 COMMUNICATION-AWARE ACQUISITION

We provide the implementation details of the communication-aware acquisition rule introduced in Section 4.3. The aligned communication feature has spatial size $H \times W$ with 128 channels. For each candidate deployment frame f, feature energy is computed as the L2 norm over the 128 channels at every valid aligned BEV position. The receiver and sender 75th-percentile energy thresholds are computed independently within the frame. The shared-activation set Ω_f, defined in Section 4.3,

Algorithm 1 Implementation of communication-aware frame acquisition

Require: Candidate frames grouped by scenario; frozen receiver and sender; frame budget $B \leq 60$

Compute pose-aligned communication features for each candidate frame f.
Compute receiver and sender feature-energy maps.
Construct the shared-activation set Ω_f using the two 75th-percentile energy thresholds.
Discard frames for which $\Omega_f = \emptyset$.
Evaluate $d_{\mathrm{QK}}(f)$ at three transformer layers and average the layer-wise scores.
Within each scenario, rank candidates by decreasing $d_{\mathrm{QK}}(f)$ and compute $\pi(f)$.
Initialize an empty ordered selection list $\mathcal{S}$.
while $|\mathcal{S}| < B$ **do**
    Collect the highest-ranked eligible candidate from each non-exhausted scenario.
    Order the collected candidates by decreasing $\pi(f)$, decreasing $d_{\mathrm{QK}}(f)$, and increasing frame ID.
    Add candidates that satisfy the within-scenario temporal gap in Eq. 23.
end while
Form $\mathcal{S}_{10}$, $\mathcal{S}_{20}$, $\mathcal{S}_{40}$, and $\mathcal{S}_{60}$ as prefixes of $\mathcal{S}$.
Request commissioning annotations for the selected frames.

Ensure: Labeled commissioning sets for optimizing C_K and C_V

retains positions whose receiver and pose-aligned sender energies both meet their respective thresholds. Frames with $\Omega_f = \emptyset$ do not yield a valid QK score and are excluded from ranking.

Equation 15 suppresses the transformer-layer index for readability. In implementation, the same QK similarity is evaluated independently at each of the three V2X-ViT layers, and the resulting layer-wise scores are averaged. Each layer contains $M = 8$ attention heads with 32-dimensional query and key vectors. Cosine similarity uses $\epsilon = 10^{-8}$ for numerical stabilization. Accordingly, $c_{\mathrm{QK}}(f)$ is the mean similarity over shared-active positions, attention heads, and the three transformer layers, while the acquisition mismatch remains

$$d_{\mathrm{QK}}(f) = 1 - c_{\mathrm{QK}}(f)$$

as defined in the main text.

For scenario normalization, let a scenario contain n valid candidate frames with a finite QK mismatch score. Candidates are sorted by decreasing $d_{\mathrm{QK}}(f)$, with increasing frame ID used to break ties. Let $\kappa(f) \in \{1, \ldots, n\}$ denote the one-based rank of frame f in this ordering. Its scenario-normalized percentile score is

$$\pi(f) = \frac{n - \kappa(f) + 1}{n}. \tag{22}$$

Hence, a larger $\pi(f)$ indicates a larger mismatch relative to the other candidates from the same scenario.

Selection proceeds in scenario-balanced rounds. At each round, every scenario with at least one eligible candidate contributes its highest-ranked remaining frame. The candidates collected in the current round are ordered by decreasing $\pi(f)$, then decreasing $d_{\mathrm{QK}}(f)$, and finally increasing frame ID. Let $\mathrm{idx}(f)$ denote the temporal index of frame f. A candidate is eligible only if

$$|\mathrm{idx}(f) - \mathrm{idx}(f')| \geq 20 \tag{23}$$

for every previously selected frame f' from the same scenario. Scenarios without an eligible candidate are skipped.

A single deterministic acquisition ordering is generated first. For a frame budget B, let $\mathcal{S}_B$ denote the first B frames in this ordering. The reported communication-aware commissioning sets therefore satisfy

$$\mathcal{S}_{10} \subset \mathcal{S}_{20} \subset \mathcal{S}_{40} \subset \mathcal{S}_{60}. \tag{24}$$

This construction ensures that larger annotation budgets extend the same selected set rather than using independently sampled subsets.

For the reported experiments, the calibration candidate pool is restricted to frames containing at least one annotated target and a valid sender observation. This restriction is used only to construct the experimental calibration pool; the QK acquisition score itself does not use detection labels.

For the acquisition baselines, **RawQK** ranks all valid candidate frames globally by decreasing $d_{\mathrm{QK}}(f)$, with increasing frame ID used to break ties, and applies no diversity constraint. **Random-diverse** is seed-dependent: it randomizes the scenario order, performs temporal maximin selection from a random starting point within each scenario, and combines candidates through scenario round-robin. Unlike the final communication-aware strategy, Random-diverse does not impose the fixed 20-frame temporal gap.

A.3.3 Calibration set construction

The unit of acquisition is a commissioning frame, and annotation cost is reported both as the number of selected frames and as the number of ground-truth boxes they contain. As described in Section A.1.1, the reported commissioning experiments reuse OPV2V training observations as an experimental proxy for paired data available after a persistent partnership is established. The base perception systems are frozen before this stage.

For the limited-budget experiments, the candidate pool contains 5,659 frames from 42 scenarios. It is restricted to frames that contain at least one annotated target, include sender B_1, and yield a valid QK mismatch score. This restriction defines the experimental calibration pool but does not enter the communication-aware score itself: once the candidate pool is formed, $d_{\mathrm{QK}}(f)$ is computed entirely from frozen communication activations and poses without using detection labels.

The final communication-aware ordering produces nested commissioning subsets containing 111, 263, 622, and 935 ground-truth boxes at 10, 20, 40, and 60 frames, respectively. The communication-aware manifest is fixed across optimization seeds, whereas Random-diverse changes its selected manifest with the seed; its box counts are therefore reported as mean and standard deviation in Table 9. Full box annotations for the selected frames are then used to optimize the CAC adapters on the commissioning subset. Commissioning checkpoint selection uses the OPV2V validation split, and final AP is measured on the disjoint OPV2V test split. Full commissioning uses all 6,374 OPV2V training frames as the calibration proxy.

A.3.4 Optimization

We use a fixed three-seed protocol for limited-budget commissioning. For NormalizedQK + diversity, the selected manifest is fixed across seeds so that only optimization randomness varies, whereas Random-diverse varies both the selected manifest and optimization randomness. Training runs for ten epochs with batch size one using Adam with learning rate 10^{-3}, $(\beta_1, \beta_2) = (0.9, 0.999)$, $\epsilon = 10^{-10}$, and weight decay 10^{-4}. After epoch five, the learning rate is multiplied by 0.1. No data augmentation is applied.

The receiver-local ground-truth objective consists of focal classification with $\alpha = 0.25$, $\gamma = 2$, and weight 1; SmoothL1 regression with $\sigma = 3$ and weight 2; and direction cross-entropy with weight 0.2. No pseudo labels are used. The checkpoint with the lowest OPV2V validation loss is selected.

During commissioning, the complete base model remains frozen in evaluation mode, including its batch-normalization and dropout layers, and only C_K and C_V are optimized. The LayerNorm and GroupNorm layers inside the adapters do not maintain running statistics. The zero-shot safeguard is disabled throughout CAC. Limited-budget Random-diverse and NormalizedQK + diversity results report the mean and standard deviation over three fixed seeds. RawQK is reported as a single run, while the full-ground-truth adapter ablation and full commissioning reference are point estimates.

A.4 Generalization across Receiver Systems

Table 1 in the main text fixes receiver A_1 and varies the unseen sender among B_1, B_2, and B_3, showing that IsoComm supports zero-shot cooperation across independently trained sender architectures. Here, we examine the complementary question of whether this behavior depends on the receiver used to prepare the communication interface. We therefore fix the unseen sender to B_1 and vary the receiver across A_1, A_3, and A_4, covering PointPillar, SECOND, and camera-based LSS–ResNet101 systems. Table 6 reports Ego-only, Direct fusion, and IsoComm for each receiver so that cooperative performance is interpreted relative to the corresponding receiver's standalone capability.

Table 6: Receiver generalization with fixed unseen sender B_1. Each result is $AP_{30}/AP_{50}/AP_{70}$ (%). The range column reports the configured x/y half-range in meters, rather than full width and height. Comparisons are within a receiver, and no cross-receiver average is computed.

Receiver	Modality / architecture	x/y half-range (m)	Ego-only	Direct fusion	IsoComm
A_1	LiDAR / PointPillar, deep BEV	102.4×51.2	84.42 / 82.49 / 70.06	57.84 / 57.66 / 53.41	90.68 / 88.46 / 74.22
A_3	LiDAR / SECOND	102.4×51.2	80.64 / 78.53 / 66.41	80.56 / 79.56 / 68.79	86.08 / 83.58 / 72.24
A_4	Camera / LSS–ResNet101	51.2×51.2	33.13 / 22.37 / 8.23	34.49 / 23.70 / 9.70	44.58 / 30.28 / 14.08

Table 7: CAC adapter design on $A_1 \leftarrow B_1$ under full-ground-truth commissioning, with clipping off. Results are point estimates in percent. The final design trains C_K and spatial full-channel C_V.

Trainable interface	Parameters	AP_{30}	AP_{50}	AP_{70}
C_K only	33,216	94.03	90.81	75.11
Full-channel C_V	33,216	95.19	91.71	75.15
Spatial full-channel C_V	107,328	95.62	92.38	75.91
C_K+ spatial full-channel C_V	140,544	**96.09**	**93.01**	**76.75**

IsoComm improves AP_{50} over Ego-only by 5.97 points for A_1, 5.05 points for A_3, and 7.91 points for A_4. The SECOND receiver improves from 78.53 to 83.58 AP_{50}, while the camera receiver improves from 22.37 to 30.28. Direct fusion behaves differently across these receivers: it substantially degrades A_1, while providing smaller gains for A_3 and A_4. IsoComm instead yields a positive cooperative gain for every evaluated receiver. Together with the sender-side variation in Table 1, these results show that the benefit of preparing the communication interface is not specific to the A_1 PointPillar receiver. It also extends to receivers with a different LiDAR architecture and a different sensing modality. Absolute scores are not compared across receivers because their native perception capabilities and spatial evaluation ranges differ.

A.5 Additional Ablations

The main text reports the final zero-shot and commissioning configurations. We provide additional ablations here to clarify two design choices underlying the persistent-partnership setting: which parts of the communication interface are calibrated by CAC, and how deployment frames are selected under a limited annotation budget.

A.5.1 CAC adapter design

Table 2 in the main text reports commissioning performance using the final C_K+ spatial full-channel C_V interface, but does not separate the contributions of the key and value calibration paths. We therefore compare alternative adapter placements to determine which communication components should be calibrated while keeping both base perception systems frozen. Table 7 evaluates key-only calibration, pointwise full-channel value calibration, spatial full-channel value calibration, and their final key-plus-spatial-value combination on $A_1 \leftarrow B_1$ under full-ground-truth commissioning. The zero-shot safeguard is disabled for all variants.

Calibrating the value path is more effective than key-only calibration in this setting: the full-channel C_V reaches 91.71 AP_{50} compared with 90.81 for C_K. Adding spatial processing further raises AP_{50} to 92.38, indicating that the value-side correction benefits from spatial context. Combining C_K with the spatial value adapter gives the strongest result across all three IoU thresholds, reaching 93.01 AP_{50} and 76.75 AP_{70}. We therefore use this joint interface for the limited-annotation commissioning results in Table 2 and the experiments below.

A.5.2 Acquisition strategy

Table 2 in the main text compares the final communication-aware selection strategy with scenario-diverse random sampling at 40- and 60-frame budgets. That comparison establishes the annotation efficiency of the complete CAC acquisition procedure, but does not show how the QK mismatch ranking behaves before scenario normalization and diversity control are applied. We therefore com-

Table 8: Acquisition strategy ablation on $A_1 \leftarrow B_1$. All methods train the final CAC interface at the stated frame budget with clipping off. AP values are in percent. RawQK is a single run; the other strategies report mean $\pm$ standard deviation over three seeds.

Frames	Selection	AP_{30}	AP_{50}	AP_{70}
10	Random-diverse	91.80 ± 0.48	89.26 ± 0.28	74.16 ± 0.26
	RawQK (single run)	90.39	88.42	73.78
	NormalizedQK + diversity	91.98 ± 0.59	89.36 ± 0.35	73.66 ± 0.52
20	Random-diverse	93.00 ± 0.76	89.82 ± 0.35	73.91 ± 0.64
	RawQK (single run)	90.56	88.65	73.65
	NormalizedQK + diversity	92.64 ± 0.17	89.81 ± 0.23	73.94 ± 0.17
40	Random-diverse	93.30 ± 0.55	90.19 ± 0.38	73.89 ± 0.64
	RawQK (single run)	91.41	89.07	73.86
	NormalizedQK + diversity	94.25 ± 0.27	90.76 ± 0.16	74.09 ± 0.19
60	Random-diverse	93.57 ± 0.30	90.44 ± 0.28	73.98 ± 0.28
	RawQK (single run)	91.37	89.00	73.34
	NormalizedQK + diversity	94.53 ± 0.11	91.01 ± 0.07	74.46 ± 0.22

pare three acquisition strategies under the same receiver–sender pair, adapter architecture, optimization protocol, and annotation budgets: Random-diverse, global RawQK ranking, and the final NormalizedQK + diversity strategy.

RawQK remains below the diversity-aware strategies across the evaluated budgets. At 40 frames, NormalizedQK + diversity reaches 90.76 ± 0.16 AP_{50}, compared with 90.19 ± 0.38 for Random-diverse and 89.07 for RawQK. At 60 frames, it reaches 91.01 ± 0.07, again exceeding both alternatives. At the smaller 10- and 20-frame budgets, Random-diverse and NormalizedQK + diversity are close, while the advantage of the final strategy becomes clearer as more frames are selected. These results support combining communication mismatch with scenario normalization and diversity control rather than using the raw mismatch ranking alone.

A.6 Commissioning under Limited Annotation

Table 2 in the main text focuses on the 40- and 60-frame operating points that highlight the annotation efficiency of CAC, together with zero-shot and full-commissioning references. To show how commissioning evolves across the full measured annotation range, we additionally report the complete budget curves here. Table 9 includes Random-diverse results from 1 to 60 frames, all measured communication-aware budgets from 10 to 60 frames, and the corresponding labeled-object counts. This provides a more complete view of how quickly the frozen zero-shot system can be commissioned as additional annotations become available.

The complete curve shows a progressive improvement from the zero-shot operating point as additional labeled deployment frames become available. Consistent with the main comparison in Table 2, the 40-frame communication-aware configuration reaches 90.76 ± 0.16 AP_{50}, exceeding the 90.44 ± 0.28 obtained by Random-diverse with 60 frames. It therefore uses 33.3% fewer labeled frames. The corresponding communication-aware subset contains 622 labeled boxes, compared with 915.7 ± 28.0 boxes for Random-diverse at 60 frames, or approximately 32.1% fewer annotated objects. Increasing the communication-aware budget to 60 frames further raises AP_{50} to 91.01 ± 0.07, while full commissioning reaches 93.01 AP_{50}.

Figure 4 visualizes the same measured AP_{50} trajectories against both frame and labeled-object budgets. The curves connect measured operating points only and do not interpolate unobserved annotation budgets. Table 9 additionally retains AP_{30} and AP_{70} and the measured box counts for the limited-budget subsets.

Table 9: Complete commissioning results on $A_1 \leftarrow B_1$. Limited-budget AP values are mean $\pm$ standard deviation over three seeds, with measured annotation counts reported for those subsets. Random-diverse box counts vary across selected manifests; the communication-aware manifest is fixed across optimization seeds. Zero-shot and full commissioning are reference operating points.

Selection	Frames	GT boxes	AP_{30}	AP_{50}	AP_{70}
Zero-shot	0	0	90.68	88.46	74.22
Random-diverse	1	11.0 ± 2.6	90.61 ± 0.07	88.47 ± 0.08	74.08 ± 0.12
	5	70.0 ± 11.8	91.17 ± 0.87	88.84 ± 0.51	73.86 ± 0.11
	10	143.7 ± 3.8	91.80 ± 0.48	89.26 ± 0.28	74.16 ± 0.26
	20	298.0 ± 30.1	93.00 ± 0.76	89.82 ± 0.35	73.91 ± 0.64
	40	633.3 ± 4.6	93.30 ± 0.55	90.19 ± 0.38	73.89 ± 0.64
	60	915.7 ± 28.0	93.57 ± 0.30	90.44 ± 0.28	73.98 ± 0.28
NormalizedQK + diversity	10	111	91.98 ± 0.59	89.36 ± 0.35	73.66 ± 0.52
	20	263	92.64 ± 0.17	89.81 ± 0.23	73.94 ± 0.17
	40	622	94.25 ± 0.27	90.76 ± 0.16	74.09 ± 0.19
	60	935	94.53 ± 0.11	91.01 ± 0.07	74.46 ± 0.22
Full commissioning	6374	–	96.09	93.01	76.75

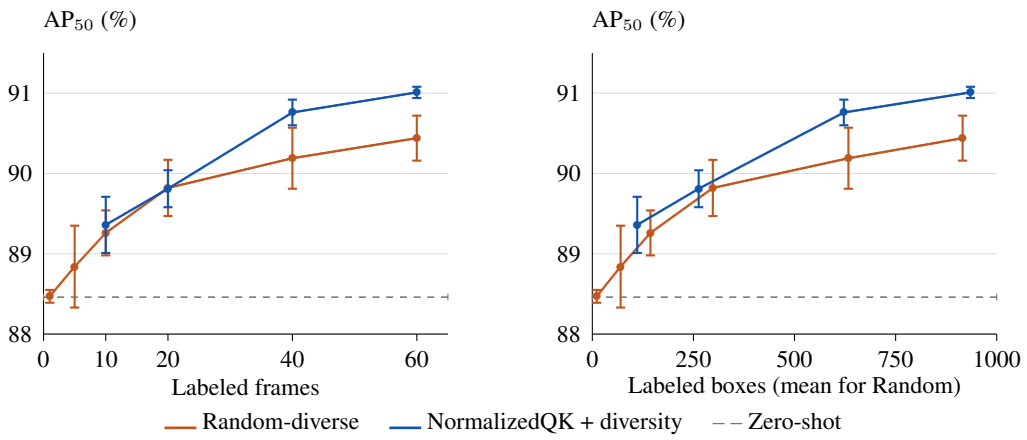

Figure 4: Measured AP_{50} commissioning curves on $A_1 \leftarrow B_1$ against labeled frames (left) and labeled boxes (right). Error bars are one standard deviation over three seeds. Horizontal positions for Random-diverse in the right panel use mean box counts; their standard deviations are reported in Table 9. Lines connect measured means as visual guides and do not denote additional evaluated budgets. The dashed reference is zero-shot IsoComm at 88.46 AP_{50}.