# Planning-oriented Cooperative Perception among Heterogeneous Vehicles

Han Zheng, Fan Ye, Yuanyuan Yang

***Abstract*—Vehicle-to-vehicle (V2V) based cooperative perception enhances autonomous driving by overcoming single-agent perception limitations such as occlusions, without relying on extensive infrastructure. However, most existing methods have two key limitations. They treat cooperative perception in isolation, with little consideration for downstream tasks such as planning, leading to poor coordination and inefficient planning decisions. They also assume perception model homogeneity across all vehicles, which can be impractical among vehicles from different manufacturers. To bridge such gaps, we propose Scout, an early-fusion framework for planning-oriented cooperative perception among vehicles of heterogeneous models. Specifically, we formalize a notion of $\Delta\theta$-*Risk Increment Distribution (RID)* to capture the distribution of the risk increment by incomplete perception to the current trajectory plan, and define a Priority Index (PI) metric for prioritizing cooperative perception on riskier regions. We develop algorithms to estimate $\Delta\theta$-*RID* and PI at run-time with theoretical bounds. Empirical results demonstrate that Scout surpasses state-of-the-art methods and strong baselines on challenging benchmarks, achieving higher success rates with only 3-10% of their communication volume.**

## I. Introduction

Cooperative perception provides comprehensive environment perception to on-road vehicles by aggregating complementary information, enhancing driving safety, efficiency, and comfort. inter-vehicle communication channel has been utilized for cooperative perception without relying on dedicated edge infrastructures, holding potential for much broader coverage at lower costs [1] [2]. Recent years have witnessed many relevant works on V2V based cooperative perception [3]–[10]. However, these approaches exhibit two limitations not yet addressed: 1) Conceptually, to perform a wide diversity of tasks and achieve advanced-level intelligence, most of them treat cooperative perception as a standalone task, with little consideration on the needs and constraints of downstream driving tasks (e.g., balancing between safety, efficiency and comfort), this leads to poor coordination thus possibly increased risks and delays in driving decisions (e.g., using much bandwidth to communicate data on less relevant objects to safety). 2) From a methodology perspective, many works exchange and aggregate intermediate features of the perception model to achieve a trade-off between perception precision and communication overhead. This implicitly assumes homogeneity of the perception model (identical feature structure) among all vehicles. However, a significant domain gap may exist in such features between heterogeneous models in vehicles from different automakers. Thus fusion of features may become difficult or impossible for multi-vehicle perception. While recent work [11] addresses this by fusing perception model outputs (e.g., bounding box proposals and confidence scores) instead of intermediate features, they require calibrators pretrained on the same standard dataset, which is still impractical among vehicles of different manufacturers.

To this end, we propose "Scout", a planning-oriented cooperative perception framework under model heterogeneity. Revisiting the early fusion strategy where vehicles fuse raw data, we recognize that it provides both high perceptual precision and is model-agnostic. While the amount of data for all areas can be large in raw data fusion, inspired by recent works on planning-oriented perception model evaluation [12] [13] [14], we realize that not all regions are equally important for driving. Fusing data from those occluded and driving-crucial regions can significantly enhance driving safety while not exceeding communication limitations. Informally, We formalize a notion of $\Delta\theta$-*Risk Increment Distribution (RID)* to capture the probability distribution of risk increment to the current trajectory plan if an occluded region remains unknown. Based on such distribution, we then define the Priority Index (PI) as a metric to prioritize data fusion for the most crucial regions, ensuring effective perception within communication constraints. We summarized our contributions as follows:

- We critically examine current cooperative perception designs for autonomous driving in light of planning orientation and heterogeneity compatibility, advocating the necessity of their considerations in cooperative perception rather than an isolated design.
- We present Scout, a risk-aware and communication-efficient framework based on early fusion. The key component is to determine driving-crucial occlusions and thus schedule cooperative perception by formalizing notions of $\Delta\theta$-*RID* and the Priority Index. We develop algorithms to estimate them at run-time, accompanied by bounds and error analysis.
- Empirical results on challenging benchmarks show that Scout outperforms state-of-the-arts (SOTAs) and strong baselines in a series of driving-related metrics, e.g., higher success rates than two SOTAs while only requiring 3-10% of their communication volume.

## II. Related Work

**V2V based cooperative perception.** Contemporary works on cooperative perception adopt early, intermediate, or late fusion strategies. Intermediate fusion methods (e.g., V2VNet [4], DiscoNet [5], AttFuse [7], Where2comm [6], CORE [9] and How2comm [15]) broadcast intermediate features from perception models, balancing perception performance and communication volume. However, these approaches assume identical models across all collaborating vehicles/infrastructures, which is unlikely among vehicles of different manufacturers, thus greatly degrading performance in fusion. Late fusion, which merges detected bounding boxes, holds the potential for addressing model heterogeneity but may result in poor perception accuracy [11]. Intermediate and late fusion do not transmit

This work was partially supported by NSF grant 2007715.

The authors are with Stony Brook University, Stony Brook, NY 11727, USA. Email: {han.zheng, fan.ye, yuanyuan.yang}@stonybrook.edu

raw perceptual data, making it difficult for the ego vehicle to re-detect objects that may have been missed. Early fusion (e.g. [1] [16]) shares raw sensor data among vehicles, provides strong perception performance and is module-agnostic, making it suitable for cooperative perception. However, sharing raw sensor data incurs large communication volumes. Scout addresses this by quantifying the risks of occluded regions to the trajectory plan and prioritizing communication on critical regions, thus greatly reducing communication overhead while preserving perception accuracy. We note that our approach is orthogonal to data compression methods, which can be incorporated to further reduce communication volume.

**Risk assessment.** Assessing scenario risk, particularly with occlusions, is critical for autonomous vehicle safety. Various approaches have offered different definitions of "risk". Hu et al. define the risky region as the unknown region so far [6]. In [17] and [18], the authors define a risky region as one that has not been observed for a long time. However, equating all occlusions oversimplifies the issue by ignoring the vehicle-environment interaction. Cui et al. utilize an end-to-end training paradigm to implicitly determine the risk areas from the supervision of the planner [2], but learning-based methods often lack Out-of-Distribution (OOD) robustness, interpretability, or guarantees. A typical family of risk assessment is the Forward Reachability Set (FRS) [19] [20], which calculates all future behaviors of other road users according to the assumptions made (e.g. constant speed and heading) to assess whether a scenario will lead to a hazardous event in the future. However, it neglects the dependency between the single-vehicle-perceived scenario and the conjectured scenario with reasonable inferences about unknowns, resulting in inaccurate risk assessment. Anatonante et al. explicitly models the dependency between the perceived and the generated plausible scenarios for identifying risky perception failures [21], though their method only qualitatively determines whether the risk has increased without quantifying the increment or the distribution of the increment. Our method explicitly quantifies the risk increment if an occluded region remains unknown and captures its distribution, providing a more comprehensive and interpretable risk assessment.

## III. $(\theta, \Delta\theta)$-Risk Increment Distribution

### A. Scenario modeling

Formally, let $\boldsymbol{s}_t \in \mathbb{S} \subseteq \mathbb{R}^n$ be an estimate of the world state at time $t$ based on the ego vehicle's perception, which comprises the state (e.g., velocity, heading and location) of the ego vehicle, states of non-ego agents (including vehicles and pedestrians), and map attributes (e.g., lane lines, stop signs and traffic signals). Conditioned on a perceived world state history $\boldsymbol{s}_{t-T_h:t}$ in a past time period $T_h$ and the ego vehicle's current trajectory plan $\iota_t^e$ at time $t$, we assume the availability of a trajectory prediction module that provides a sequence of distribution $\{\psi(\boldsymbol{s}_{t:t+\tau}|\boldsymbol{s}_{t-T_h:t}, \iota_t^e)\}_{\tau=1}^{T_f}$ over the world states in a future time period $T_f$. Scout only assumes the availability of a module meeting such criteria but is not tied to specific implementations. We denote $c : \mathbb{S} \to \mathbb{R}^+$ to be the cost function mapping a world state to a real value, where higher values indicate riskier scenarios for the ego vehicle. We adopt a cost function that is roughly inverse-proportional to the estimated minimal time to collision with other agents (see Appendix B). Other metrics can also be adapted as cost functions [22]. The future world state distributions yield a sequence of cost distributions $\{\phi(c_{t:t+\tau}|\boldsymbol{s}_{t-T_h:t}, \iota_t^e)\}_{\tau=1}^{T_f}$ over the costs in the future. For simplicity, we focus on the cost distribution at a particular time and denote it as $\phi(c)$.

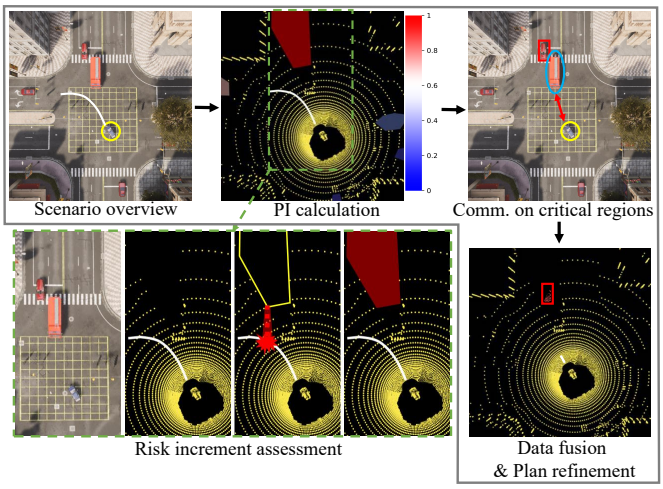



Figure 1. (Better viewed in color.) **Scenario overview**: one ego vehicle (circled in yellow) encounters several occluded regions within its perceptual field during driving. **PI calculation**: the ego vehicle uses its onboard perception system to obtain the point cloud (yellow dash lines) and compute an initial trajectory plan (the white line); then it assesses for each of the occluded areas a Priority Index (PI), which quantifies the risks by potential unseen objects in that area. Regions with higher PIs indicate a greater potential to impact the ego vehicle's safety. **Comm. on critical regions**: With a constrained communication bandwidth, the ego vehicle selectively requests raw sensor data (e.g., point clouds) for the regions with the highest PIs. **Data fusion & Plan refinement**: the ego vehicle integrates such data with its own sensor inputs to form a more complete understanding of its environment and adjusts its trajectory plan in response to the newly obtained information, thereby improving decision-making in complex environments.

### B. $(\theta, \Delta\theta)$-Risk Increment Distribution Formalization

Understanding the risk increment the vehicle would encounter if an occluded region remains unknown is the foundation of the cooperative perception in Scout. To achieve this, we treat each occluded region individually and conjecture phantom agents within them. Analyzing all regions is often computationally intensive and unnecessary since collision risks of a region are typically accounted for by its boundaries [23] [24]. Inspired by edge-based FRS methods [24] [25], we consider lane information and spawn phantom agents of the "riskiest" cases: at the edge(s) closest to the ego vehicle's trajectory plan, and seek to collide with that planned trajectory in the shortest time. Gaussian noises are added to the agents' velocity magnitudes and directions. After conjecturing the phantom agent, we calculate the cost over the state of the conjectured world. Let $\hat{\boldsymbol{s}} \in \mathbb{S} \subseteq \mathbb{R}^n$ be the world state with such conjectured phantom agents in an occluded region. We denote the distribution over the conjectured world state by $\hat{\psi}(\hat{\boldsymbol{s}})$, which induces another predicted cost distribution denoted by $\hat{\phi}(c)$. Given the two cost distributions over the two world, we formalize $\Delta\theta$-*RID* as follows:

**Definition III.1** ($\Delta\theta$-Risk Increment Distribution)**.** *Let $\phi := \phi(c)$ and $\hat{\phi} := \hat{\phi}(c)$ be the cost distribution over the perceived world and conjectured world, respectively. Let $\theta \in \mathbb{R}^+$ be the*

*cost threshold the planner desires to stay below. We then define the* $\Delta\theta$-Risk Increment Distribution (RID) *as:*

$$RID(\Delta\theta) = Pr_{c\sim\phi,\hat{c}\sim\hat{\phi}}(\hat{c} > \theta + \Delta\theta | c \leq \theta). \quad (1)$$

We note that the notion of $\Delta\theta$-*RID* does not just provide a risk increment value, but captures a probability distribution in terms of risk increment, offering a more comprehensive understanding of the risk from an occluded area. We illustrate the definition of $\Delta\theta$-*RID* in Figure 2. The choice of $\theta$ defines a desired safety threshold, which is determined by the particular onboard planner and what kind of scenario (e.g., city or highway) the vehicle is. For example, a conservative planner may tend to keep far away from other vehicles leading to an overall low $\theta$. Even for the same planner, the $\theta$ can be different when the vehicle drives on different scenarios. It can be difficult to set a proper $\theta$ directly. Insighted by [21], we use a sampling method to compute $\Phi$ and simply set $\theta$ as $\Phi^{-1}(p)$, the $p$-quantile of the current sampled $\Phi^{-1}$, which is the reverse of the marginal cumulative distribution function (CDF) of $\phi$. The definition 1 can be rewritten as:

$$RID(\Delta\theta) = Pr_{c\sim\phi,\hat{c}\sim\hat{\phi}}(\hat{c} > \Phi^{-1}(p) + \Delta\theta | c \leq \Phi^{-1}(p)). \quad (2)$$

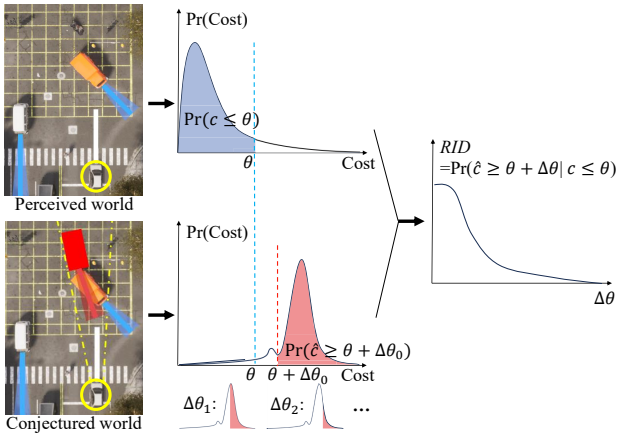



Figure 2. (Better viewed in color.) Illustration of $\Delta\theta$-*RID*. The ego vehicle's initial trajectory plan is represented in white, and the trajectory predictions of other vehicles are in blue (the uncertainty is represented in a semi-transparent form). Scout conjectures a phantom vehicle (the red rectangle) in the region occluded by the truck, which leads to a shift of cost distribution. Given a $\theta$, $\Delta\theta$-RID measures the conditional probability that the cost/risk in the conjectured world increases at least $\Delta\theta$, describing the distribution over the accumulated probability in terms of the cost increase in the conjectured world if the vehicle follows the current trajectory plan.

### C. $\Delta\theta$-Risk Increment Distribution Estimation

We note that the distributions $\phi$ and $\hat{\phi}$ are dependent, without explicit analytical representations for them or their CDFs $\Phi$ and $\hat{\Phi}$. To address this, we sample from these distributions independently for a $\Delta\theta$-RID estimation. Specifically, let $\{c_i\}_{i=1}^n \sim \phi$ and $\{\hat{c}_i\}_{i=1}^n \sim \hat{\phi}$ be i.i.d. samples from $\phi$ and $\hat{\phi}$, respectively. The empirical estimates of $\Phi$ and $\hat{\Phi}$ are computed by $\Phi^{(n)}(\cdot) \leftarrow \frac{1}{n}\sum_{i=1}^n \mathbf{1}[c_i \leq \cdot]$ and $\hat{\Phi}^{(n)}(\cdot) \leftarrow \frac{1}{n}\sum_{i=1}^n \mathbf{1}[\hat{c}_i \leq \cdot]$, where $\mathbf{1}$ is the indicator function. With the estimates by sampling, we have Probably Approximately Correct (PAC) bounds for the $\Delta\theta$-*RID* as follows:

**Theorem III.1** (Bounds on $\Delta\theta$-RID)**.** *Let* $\alpha \in (0,1)$. *With probability* $1-\alpha$*:*

$$RID(\Delta\theta) \geq 1 - min\{1, \frac{\overline{\hat{\Phi}}(\underline{\Phi^{-1}}(p) + \Delta\theta)}{p}\}, \quad (3)$$

$$RID(\Delta\theta) \leq 1 + \max\{0, \frac{\underline{\hat{\Phi}}(\overline{\Phi^{-1}}(p) + \Delta\theta) + p - 1}{p}\}, \quad (4)$$

*where*

$$\underline{\hat{\Phi}}(\cdot) = [\hat{\Phi}^{(n)} - \epsilon(n,\alpha)](\cdot), \underline{\Phi^{-1}}(\cdot) = [\Phi^{(n)} - \epsilon(n,\alpha)]^{-1}(\cdot),$$
$$\overline{\hat{\Phi}}(\cdot) = [\hat{\Phi}^{(n)} + \epsilon(n,\alpha)](\cdot), \overline{\Phi^{-1}}(\cdot) = [\Phi^{(n)} + \epsilon(n,\alpha)]^{-1}(\cdot),$$
$$\epsilon(n,\alpha) = \sqrt{ln(2/\alpha)/(2n)}.$$

*Proof: See Appendix A.*

Informally, we use the Bayesian theorem to rewrite the definition and copula [26] to model the joint distribution of the dependent distributions. It was pointed out that copulas may be used to construct multivariate distribution functions from univariate ones. Furthermore, we use the Dvoretzky–Kiefer–Wolfowitz Confidence Interval [27] [28] along with the monotonicity of functions to bound the empirical estimates with empirical samples.

## IV. Risk-Increment-aware Cooperative Perception

### A. Priority Index

We define a Priority Index (PI) based on the risk increment analysis to determine riskier regions for driving as follows:

**Definition IV.1** (Priority Index)**.** *For a given cost quantile* $p$*, let* $RID(\Delta\theta|p)$ *be the conditioned risk increment distribution of an occluded region. We define the Priority Index as:*

$$PI = \int_0^{\theta_{max} - \Phi^{-1}(p)} RID(\Delta\theta|p) d(\Delta\theta), \quad (5)$$

where $\theta_{max}$ is the maximum value of cost. Equation 5 calculates the area under the curve $RID(\Delta\theta|p)$, which implies the expectation of cost increment $\Delta\theta$. Consequently, a higher PI corresponds to a greater expected risk increment, indicating that the associated occluded region should be prioritized for perception. This prioritization helps mitigate potential risks by enhancing the vehicle's situational awareness. This integration cannot be computed analytically. We estimate the PI with Gaussian Quadrature as follows:

**Theorem IV.1** (Bounds on Priority Index)**.** *Let* $p$ *be the given cost quantile. Assume* $\Phi$ *and* $\hat{\Phi}$ *are locally smooth. Let* $n_{\Delta\theta}$ *be the number of samples of* $\Delta\theta$*. For each* $\Delta\theta$*, let* $n$ *be the number of samples on both* $\phi$ *and* $\hat{\phi}$*. Let* $\alpha \in (0,1)$*. With probability* $1-\alpha$*:*

$$PI > \sum_{i=1}^{n_{\Delta\theta}} w_i g(\Delta\theta_i) - \nu,$$

*where*

$$g(\Delta\theta) = 1 - min\{1, \frac{\overline{\hat{\Phi}}(\underline{\Phi^{-1}}(p) + \Delta\theta)}{p}\}.$$

*the nodes (the sampled values of* $\Delta\theta$*)* $\{\Delta\theta_i\}_{i=1}^{n_{\Delta\theta}}$*, the weights* $\{w_i\}_{i=1}^{n_{\Delta\theta}}$*, and the error* $\nu$ *are determined by the particular numerical integration method used.*

*See Appendix C for proof and particular expressions of $\nu$ and $\lambda$ when using the Golub-Welsch algorithm [29].*

We utilize the above lower bound of PI to determine the regions to prioritize in communication, as it reflects the minimal risk level associated with a region. The error of such estimation is quantified by $\epsilon$ in Theorem III.1 and $\nu$ above. Increasing the number of samples ($n$ and $n_{\Delta\theta}$) reduces the error, resulting in more accurate estimations, but it also poses a higher computational overhead. We empirically show the sample size chosen achieves an effective balance between accuracy and computational efficiency.

We also point out that as the state of the agent changes continuously and the cost function used is locally smooth, thus the local smoothness assumption on the CDFs of cost distributions is reasonable. The computation complexity of the integration comprises of a $O(n_{\Delta\theta}^2)$ from $\Delta\theta$ sampling using the Golub-Welsch algorithm and a $O(n)$ from cost sampling, where the former can be replaced with other numerical integration methods (e.g. Simpson's [30]) for a different trade-off between accuracy and complexity. In our context, the sampling calculates the time to collision given the two trajectories, which is simple and computationally tractable. We further empirically show the computation time in Section V-C.

### B. Risk Increment-aware Early Fusion

We implement an early fusion strategy by fusing raw point cloud data for cooperative perception. The ego vehicle computes Priority Indices (PIs) for occluded regions and requests point clouds for regions by descending order of PI until the given bandwidth is used up. We assume vehicles share a common coordinate system and know each other's positions and headings like [6]. The ego vehicle aligns the received point clouds with its own perception using relative position and heading information. Existing methods for handling temporal asynchrony and spatial error in data fusion can be integrated into Scout for improved performance (e.g., [31] and [32]). We point out that Scout is compatible with a wide range of autonomy modules (e.g., learning-based and analytical planners), as it reuses outputs from common onboard systems without relying on specific model structures or parameters, allowing it to function effectively among heterogeneous vehicles with different modules.

## V. Experiments

### A. Experimental Settings

**Scenario Configuration.** We focus on and evaluate our approach with safety-critical scenarios. We evaluate approaches using the publicly available dataset AutoCastSim [2], including *Overtaking*, *Unprotected left turn*, and *Red-light violation* scenarios. This focus is informed by data showing a higher frequency of accidents in such situations, with a notable prevalence of personal injuries occurring in urban settings.

**Metrics.** Considering the three main concerns for cooperative autonomous driving (driving safety, efficiency, and communication), we report three metrics respectively, including Success Rate (SR), Success weighted by Completion Time (SCT) [1] [2], and Communication Volume.

[1]SCT measures the success rate as the ratio of completion time between the omniscient expert and the ego car. Since the expert should not take longer to complete than the ego car, SCT lies in the range [0, 1], with higher values indicating more efficient driving.

**Implementation Details.** We conduct our experiments on a workstation with an Intel i9 CPU and NVIDIA GeForce RTX 3090 GPU. For modules in the autonomy stack, we use PredictionNet [33] for perception as well as trajectory prediction and PDM-closed [34] for planning by default. We assign each vehicle a perception range of 140m [9] and a communication range of 170m. We simulate an LTE-Direct QPSK with 10 MHz bandwidth for V-V communication [35], which translates to a peak rate of 7.2 Mbps. We set the maximum decision interval to 0.5s, which requires a communication volume not greater than 450 kilobytes (around $2^{18.78}$ bytes). For predefined parameters in Scout, we set the cost quantile $p = 0.99$, the confidence $1-\alpha = 0.9$, $n = 1000$ samples from both perceived and conjecture world and a maximum of node number $n_{\Delta\theta} = 5$ in PI calculation.

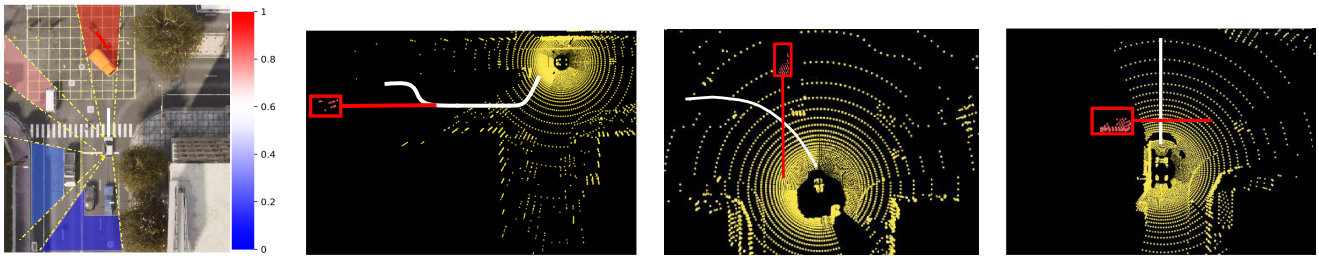
### B. Baselines

**Single**: The ego vehicle drives solely based on the onboard data and modules without cooperative perception. **Early fusion**: Ego vehicle requests all point clouds from neighbor vehicles, which exceeds the available bandwidth. To allow performance comparison, we pause the clock of the simulator until the data transmission is complete. **Simplified Reachability Quantification (SRQ) [24]**: We adapt this original work for cooperative perception, where the ego vehicle requests point clouds of desired occluded regions based on the reachability analysis of SRQ. **Where2comm [6]**: Where2comm is a SOTA approach using intermediate fusion, which prioritizes unknown areas via generating spatial confidential maps. **CORE [9]**: CORE is a SOTA approach using intermediate fusion, which studies the V-V cooperative perception from the reconstruction perspective. **How2comm [15]**: How2comm is another SOTA approach using intermediate fusion, proposing a dedicated transformer against localization errors and feature discrepancies.

### C. Main Results

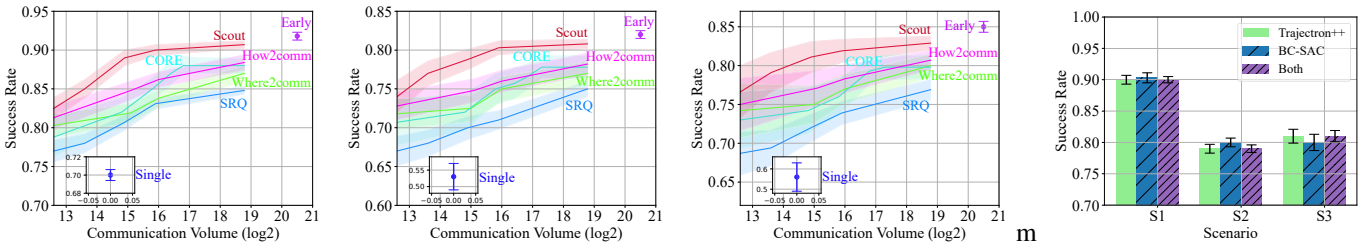
**Run time breakdown.** We first empirically study the time complexity of running Scout algorithms on our workstation. We point out that these numbers are loose higher bounds—powerful hundreds TOPS-level computers (e.g., [36]) on real autonomous vehicles can further cut down the run time to a fraction. This shows the algorithms are suitable for real-time cooperative perception among on-road vehicles. The total average time is 0.36s, highlighting the computational efficiency of our proposed algorithms. Sampling states for the perceived and conjectured worlds are negligible, taking only 0.05s and 0.01s on average (about 15% of the total run time). Priority index computation averages 0.30s, with potential for improvement through lightweight integration methods. Additionally, our approach can be easily parallelized, as PI computation for different occluded regions can be processed simultaneously, which can further accelerate Scout with the suggested optimizations.

**SR vs Comm. volume.** Figure 4(a)-4(c) show the results of success rate (average and standard deviation) with different communication volumes. We observe that compared to Single, Scout achieves much higher (more than 20%) average success rates. Scout outperforms SRQ by approximately 6% in average success rate with the same communication overhead across all scenarios, due to its superior ability to capture risk distribution and dependencies between vehicles. Additionally, Scout surpasses three state-of-the-art methods while requiring only



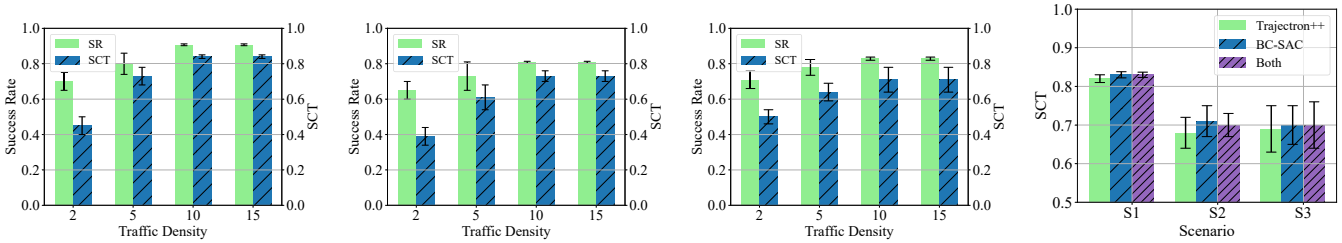
(a) PI illustration (b) Overtaking (c) Unprotected left turn (d) Red-light violation

Figure 3. (Better viewed in color.) Figure 3(a) illustrates the Priority Indexes (PI) of different occluded regions in an evaluation. The more red colors imply higher PI values. There is a vehicle in the upper region behind the truck, and Scout successfully assigns this occlusion with the highest PI value because such occluded agents pose the highest risks and could greatly impact the ego vehicle's safety. Figure 3(b)-3(d) illustrate the data fusion in three scenarios, with the point clouds (yellow dashed lines) perceived by the ego vehicle and the initial trajectory plans (white lines). The occluded vehicles are represented by red bounding boxes, and their intended trajectories are represented by red lines. With Scout, the ego vehicle receives the point clouds of occluded vehicles (represented by red dots), thus being able to detect the presence of previously occluded vehicles and adjust its plan to avoid potential collisions.




(a) Overtaking (b) Unprotected left turn (c) Red-light violation (d) Performance with different modules (SR)

Figure 4. Figure 4(a)-4(c) show the results of Success Rate vs. Communication volume in three scenarios. The communication volume results are presented in logarithmic scale base 2, and the size is counted in bytes. Figure 4(d) shows the success rate when using different modules.




(a) Overtaking (b) Unprotected left turn (c) Red-light violation (d) Performance with different modules (SCT)

Figure 5. Figure 5(a)-5(c) shows the performance under different traffic densities. We note that Scout becomes Single when density is zero, as no other vehicles provide occlusion information. Figure 5(d) shows the SCT when using different modules.

at least 1/32 of their communication volumes, respectively, by prioritizing regions critical to driving success. It achieves nearly the same success rate as early fusion but with only 1/16 of the communication volume by focusing on high-risk, occluded regions and avoiding the redundant data that early fusion indiscriminately includes.

**SCT.** We further study the average Success weighted by Completion Time (SCT) of different approaches in three scenarios, showing results in Table I[2]. We observe that the SCT of Single is significantly lower than cooperative perception-based approaches (e.g., up to 37% lower than Scout), as the ego vehicle acts more conservatively without information from other vehicles. Scout outperforms SRQ, Where2comm, CORE, and How2comm in SCT by up to 13% across cases, due to its ability to prioritize communication on critical occluded regions, leading to more accurate and faster decision-making. Scout also approaches the idealistic but impractical early fusion, which exchanges all point cloud data without any bandwidth limitation: the same SCT in scenario 1 and marginally lower performance (within a 2% difference) in scenarios 2 and 3. Given Scout's significantly reduced communication overhead, it is evident that Scout efficiently identifies the critical regions for driving and satisfies practical constraints, making it a viable solution for bandwidth-limited environments.

**Sensitivity to traffic density and different modules.** We evaluate Scout under different traffic densities (10 by default) across three scenarios (Figure 5(a)-5(c)). We found that Scout performs well in the most dense setting with 0.92 SR and 0.83 SCT on average. We also highlight that Scout is compatible with a wide range of modules in a plug-and-play

[2]In the table, SCT1-3 refers to the three scenarios.

Table I
SUCCESS WEIGHTED BY COMPLETION TIME (SCT).

| Approach | SCT1↑ | SCT2↑ | SCT3↑ |
|---|---|---|---|
| Single | 0.47±0.01 | 0.42±0.05 | 0.53±0.14 |
| Early fusion | 0.84±0.05 | 0.75±0.08 | 0.73±0.09 |
| SRQ | 0.71±0.03 | 0.60±0.05 | 0.61±0.10 |
| Where2comm | 0.76±0.04 | 0.64±0.07 | 0.64±0.11 |
| CORE | 0.77±0.05 | 0.66±0.06 | 0.65±0.09 |
| How2comm | 0.78±0.05 | 0.64±0.09 | 0.67±0.08 |
| Scout (Ours) | 0.84±0.01 | 0.73±0.03 | 0.71±0.06 |

manner, enabling it to function effectively among vehicles of heterogeneous perception modules. To validate this, we re-evaluated Scout's SR and SCT by replacing key modules, including PredictionNet with Trajectron++ [37], PDM-Closed with BC-SAC [38], and both together. The results (Figure 5(d) and 4(d)) show only slight fluctuations within $\pm 2\%$, demonstrating that Scout maintains performance with different autonomy modules.

## VI. DISCUSSIONS

Precisely sensing, modeling, and predicting the world surrounding a vehicle remain open research questions. In this work, we assume that the future states of other agents can be inferred from their past behaviors, and accordingly integrate existing trajectory prediction modules. It is notable that, while Scout is designed with early-fusion perception scheme, the risk assessment part can be adapted to many other tasks, such as driving risk alert, intermediate/late-fusion perception, and planner designs. We focus on and evaluate Scout with three particular scenarios. We are interested in studying driving risks in more general long-horizon scenes in the future.

## VII. CONCLUSION

We propose an early-fusion framework for cooperative perception among heterogeneous vehicles, emphasizing a planning-oriented approach to reduce communication overhead. We introduce a risk distribution notion for occluded regions relative to the current trajectory and a priority index metric to optimize V-V communication under bandwidth constraints, along with theoretical guaranteed algorithms for estimation. Empirical evaluations show that our framework outperforms strong baselines and state-of-the-art methods in driving safety, efficiency, and communication overhead.

## APPENDIX

### A. Proof of Theorem III.1

We use "copula" [26] [21] to model the dependencies between dependent distributions, which behaves like a joint distribution function. We present Sklar's theorem [39] to provide the theoretical foundation for applying copulas:

**Theorem A.1** (Sklar's theorem [39]). *Let $\Phi(x_1, ..., x_d)$ be a joint distribution function, and let $\Phi_i, (i = 1, ..., d)$ be the marginal distributions. Then, there exists a copula $C : [0, 1]^d \to [0, 1]$ such that $\forall x_i \in [-\infty, +\infty]$:*

$$\Phi(x_1, ..., x_d) = C(\Phi_1(x_1), ..., \Phi_d(x_d)). \tag{6}$$

Using the Bayesian theorem and Sklar's theorem, we rewrite the definition in Equation (2) as:

$$\begin{aligned} RID(\Delta\theta) &= 1 - Pr(\hat{c} \le \Phi^{-1}(p) + \Delta\theta | c \le \Phi^{-1}(p)) \\ &= 1 - \frac{C(p, \hat{\Phi}(\Phi^{-1}(p) + \Delta\theta))}{p}, \end{aligned} \tag{7}$$

where $C(\cdot, \cdot)$ is a copula. After applying the Frechet–Hoeffding Theorem [26] to the copula in Equation (7), we obtain the Inequalities (3) and (4).

Recall that we empirically estimate the CDFs $\Phi$ and $\hat{\Phi}$. Considering the non-decreasing characteristics of the CDFs and their empirical estimates, We apply the Dvoretzky–Kiefer–Wolfowitz Confidence Interval [27] [28] to $\hat{\Phi}(\Phi^{-1}(p) + \Delta\theta)$, which bounds the empirical estimates. With probability at least $1 - \alpha$ and $\epsilon(n, \alpha) = \sqrt{ln(2/\alpha)/2n}$,

$$\begin{aligned} &\hat{\Phi}(\Phi^{-1}(p) + \Delta\theta) \ge \hat{\Phi}^{(n)}(\Phi^{-1}(p) + \Delta\theta) - \epsilon(n, \alpha) \\ &\ge \hat{\Phi}^{(n)}([\Phi^{(n)} + \epsilon(n, \alpha)]^{-1}(p) + \Delta\theta) - \epsilon(n, \alpha), \end{aligned} \tag{8}$$

$$\begin{aligned} &\hat{\Phi}(\Phi^{-1}(p) + \Delta\theta) \le \hat{\Phi}(\Phi^{-1}(p) + \Delta\theta) + \epsilon(n, \alpha) \\ &\le \hat{\Phi}([\Phi^{(n)} - \epsilon(n, \alpha)]^{-1}(p) + \Delta\theta) + \epsilon(n, \alpha). \end{aligned} \tag{9}$$

Finally, applying Inequality (8) to Inequality (4) and applying Inequality (9) to Inequality (3), we complete the proof.

### B. Cost function

We define the cost function as:

$$c = 1 - min_{a \in \mathcal{A}}\{min\{1, \frac{T}{T_m}\}\}, \tag{10}$$

where $\mathcal{A}$ is the set of agents (including phantom ones), $T$ is the time until a collision between the ego vehicle and an agent and $T_m$ is a pre-defined maximum time. This cost function takes values in $[0, 1]$, where higher values indicate smaller time-to-collisions. In our experiment, we set $T_m = 3$s.

### C. Proof of Theorem IV.1

We apply Theorem III.1 to Definition IV.1 such that with probability $1 - \alpha$:

$$\begin{aligned} PI &\ge \int_0^{\theta_{max} - \overline{\Phi^{-1}}(p)} g(\Delta\theta) d(\Delta\theta) \\ &> \int_0^{\theta_{max} - \overline{\Phi^{-1}}(p)} g(\Delta\theta) d(\Delta\theta). \end{aligned} \tag{11}$$

Then we apply Gaussian Quadrature (the Golub-Welsch algorithm [29] specifically) to estimate the integration numerically:

$$\left| \int_0^{\theta_{max} - \overline{\Phi^{-1}}(p)} g(\Delta\theta) d(\Delta\theta) - \sum_{i=1}^{n_{\Delta\theta}} w_i g(\Delta\theta_i) \right| <= \nu, \tag{12}$$

where

$$\begin{aligned} \nu &= \lambda \mathbf{max}_{\xi \in \{0, \theta_{max} - \overline{\Phi^{-1}}(p)\}} |g^{(2n\Delta\theta)}(\xi)|, \\ \lambda &= \frac{(\theta_{max} - \overline{\Phi^{-1}}(p))^{2n_{\Delta\theta}+1} (n_{\Delta\theta}!)^4}{(2n_{\Delta\theta} + 1)[(2n_{\Delta\theta})!]^3}. \end{aligned}$$

For brevity, we omit the expressions of the nodes $\{\Delta\theta_i\}_{i=1}^{n_{\Delta\theta}}$ and weights $\{w_i\}_{i=1}^{n_{\Delta\theta}}$. Applying the Inequality (12) to the Inequality (11), we complete the proof.

## REFERENCES


[1] Q. Chen, S. Tang, Q. Yang, and S. Fu, "Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 514–524.

[2] J. Cui, H. Qiu, D. Chen, P. Stone, and Y. Zhu, "Coopernaut: End-to-end driving with cooperative perception for networked vehicles," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 252–17 262.

[3] Q. Chen, X. Ma, S. Tang, J. Guo, Q. Yang, and S. Fu, "F-cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3d point clouds," in *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, 2019, pp. 88–100.

[4] T.-H. Wang, S. Manivasagam, M. Liang, B. Yang, W. Zeng, and R. Urtasun, "V2vnet: Vehicle-to-vehicle communication for joint perception and prediction," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer, 2020, pp. 605–621.

[5] Y. Li, S. Ren, P. Wu, S. Chen, C. Feng, and W. Zhang, "Learning distilled collaboration graph for multi-agent perception," *Advances in Neural Information Processing Systems*, vol. 34, pp. 29 541–29 552, 2021.

[6] Y. Hu, S. Fang, Z. Lei, Y. Zhong, and S. Chen, "Where2comm: Communication-efficient collaborative perception via spatial confidence maps," *Advances in neural information processing systems*, vol. 35, pp. 4874–4886, 2022.

[7] R. Xu, H. Xiang, X. Xia, X. Han, J. Li, and J. Ma, "Opv2v: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2583–2589.

[8] Z. Chen, Y. Shi, and J. Jia, "Transiff: An instance-level feature fusion framework for vehicle-infrastructure cooperative 3d detection with transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 18 205–18 214.

[9] B. Wang, L. Zhang, Z. Wang, Y. Zhao, and T. Zhou, "Core: Cooperative reconstruction for multi-agent perception," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 8710–8720.

[10] Y. Lu, Q. Li, B. Liu, M. Dianati, C. Feng, S. Chen, and Y. Wang, "Robust collaborative 3d object detection in presence of pose errors," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 4812–4818.

[11] R. Xu, W. Chen, H. Xiang, X. Xia, L. Liu, and J. Ma, "Model-agnostic multi-agent perception framework," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 1471–1478.

[12] J. Philion, A. Kar, and S. Fidler, "Learning to evaluate perception models using planner-centric metrics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 055–14 064.

[13] B. Ivanovic and M. Pavone, "Injecting planning-awareness into prediction and detection evaluation," in *2022 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2022, pp. 821–828.

[14] W. Li and X. Yang, "Transcendental idealism of planner: Evaluating perception from planning perspective for autonomous driving," in *International Conference on Machine Learning*. PMLR, 2023, pp. 20 253–20 275.

[15] D. Yang, K. Yang, Y. Wang, J. Liu, Z. Xu, R. Yin, P. Zhai, and L. Zhang, "How2comm: Communication-efficient and collaboration-pragmatic multi-agent perception," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[16] H. Qiu, P. Huang, N. Asavisanu, X. Liu, K. Psounis, and R. Govindan, "Autocast: Scalable infrastructure-less cooperative perception for distributed collaborative driving," *arXiv preprint arXiv:2112.14947*, 2021.

[17] J. M. G. Sánchez, T. Nyberg, C. Pek, J. Tumova, and M. Törngren, "Foresee the unseen: Sequential reasoning about hidden obstacles for safe driving," in *2022 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2022, pp. 255–264.

[18] C. van der Ploeg, T. Nyberg, J. M. G. Sánchez, E. Silvas, and N. van de Wouw, "Overcoming fear of the unknown: Occlusion-aware model-predictive planning for automated vehicles using risk fields," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[19] R. Ramakrishnan, E. Kamar, B. Nushi, D. Dey, J. Shah, and E. Horvitz, "Overcoming blind spots in the real world: Leveraging complementary abilities for joint execution," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 6137–6145.

[20] R. Trauth, K. Moller, and J. Betz, "Toward safer autonomous vehicles: Occlusion-aware trajectory planning to minimize risky behavior," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 4, pp. 929–942, 2023.

[21] P. Antonante, S. Veer, K. Leung, X. Weng, L. Carlone, and M. Pavone, "Task-aware risk estimation of perception failures for autonomous vehicles," *arXiv preprint arXiv:2305.01870*, 2023.

[22] L. Westhofen, C. Neurohr, T. Koopmann, M. Butz, B. Schütt, F. Utesch, B. Neurohr, C. Gutenkunst, and E. Böde, "Criticality metrics for automated driving: A review and suitability analysis of the state of the art," *Archives of Computational Methods in Engineering*, vol. 30, no. 1, pp. 1–35, 2023.

[23] M. Koschi and M. Althoff, "Set-based prediction of traffic participants considering occlusions and traffic rules," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 2, pp. 249–265, 2020.

[24] H. Park, J. Choi, H. Chin, S.-H. Lee, and D. Baek, "Occlusion-aware risk assessment and driving strategy for autonomous vehicles using simplified reachability quantification," *IEEE Robotics and Automation Letters*, 2023.

[25] P. F. Orzechowski, A. Meyer, and M. Lauer, "Tackling occlusions & limited sensor range with set-based safety verification," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 1729–1736.

[26] R. B. Nelsen, *An introduction to copulas*. Springer, 2006.

[27] W. J. Dvoretzky A, Kiefer J, "Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator," in *The Annals of Mathematical Statistics*, 1956, pp. 642–669.

[28] M. P, "The tight constant in the dvoretzkykiefer-wolfowitz inequality," in *The Annals of Probability*, 1990, pp. 1269–1283.

[29] G. H. Golub and J. H. Welsch, "Calculation of gauss quadrature rules," *Mathematics of computation*, vol. 23, no. 106, pp. 221–230, 1969.

[30] K. Atkinson, *An introduction to numerical analysis*. John wiley & sons, 1991.

[31] P. Gao, B. Reily, R. Guo, H. Lu, Q. Zhu, and H. Zhang, "Asynchronous collaborative localization by integrating spatiotemporal graph learning with model-based estimation," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 1695–1701.

[32] Q. Zhang, X. Zhang, R. Zhu, F. Bai, M. Naserian, and Z. M. Mao, "Robust real-time multi-vehicle collaboration on asynchronous sensors," in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, 2023, pp. 1–15.

[33] A. Kamenev, L. Wang, O. B. Bohan, I. Kulkarni, B. Kartal, A. Molchanov, S. Birchfield, D. Nistér, and N. Smolyanskiy, "Predictionnet: Real-time joint probabilistic traffic prediction for planning, control, and simulation," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 8936–8942.

[34] D. Dauner, M. Hallgarten, A. Geiger, and K. Chitta, "Parting with misconceptions about learning-based vehicle motion planning," in *Conference on Robot Learning*. PMLR, 2023, pp. 1268–1281.

[35] L. Gallo and J. Härri, "Short paper: A lte-direct broadcast mechanism for periodic vehicular safety communications," in *2013 IEEE Vehicular Networking Conference*. IEEE, 2013, pp. 166–169.

[36] "In-vehicle computing for ai-defined cars." [Online]. Available: https://www.nvidia.com/en-us/self-driving-cars/in-vehicle-computing/

[37] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*. Springer, 2020, pp. 683–700.

[38] Y. Lu, J. Fu, G. Tucker, X. Pan, E. Bronstein, R. Roelofs, B. Sapp, B. White, A. Faust, S. Whiteson *et al.*, "Imitation is not enough: Robustifying imitation with reinforcement learning for challenging driving scenarios," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 7553–7560.

[39] M. Sklar, "Fonctions de répartition à n dimensions et leurs marges," in *Annales de l'ISUP*, 1959, pp. 229–231.