Appendix

A. Additional implementation details

As mentioned in section 4.2, we adopted multi-scale feature strategy to aggregate intermediate features at multiple spatial resolutions. we detail the implementation of both the multi-scale and single-scale designs. For the multi-scale strategy, we use the deformable attention to directly aggregate the multi-scale features from a ResNet backbone, as illustrated in Fig. 11. Conversely, the single-scale strategy follows a two-step process. First, a feature pyramid network (FPN) is utilized to aggregate multi-scale features. Subsequently, deformable attention is applied to aggregate information at a single spatial resolution as illustrated in Fig. 12. The current implementation leverages the multi-scale strategy for its advantages (an ablation study comparing multi-scale and single-scale fusion strategies is given in section C).

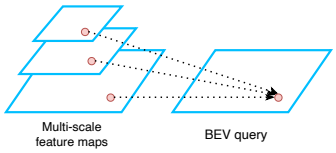

Figure 11. Multi-scale strategy

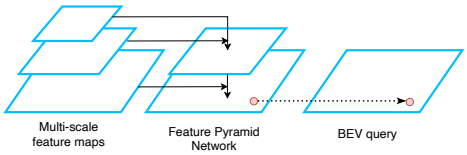

Figure 12. Single-scale strategy

B. Performance for objects with different speeds

We further investigate the compensation performance of LRCP for objects with different speeds, as shown in Tab. 3. We see that fast-moving objects are significantly more susceptible to latency compared to slower objects. LRCP effectively mitigates the effects of latency for objects across a range of speeds. Nonetheless, in extreme scenarios with both high latency and high-speed objects, the compensation performance of LRCP begins to deteriorate.

C. Ablation Studies for the main architecture

In LRCP, we use deformable attention to aggregate features from multiple agents, which achieves state-of-the-art performance on two datasets when the latency is zero. We provide ablation studies for our implementation options. Tab. 4 shows the effect of the number of self/cross attention layers in the feature fusion decoder. Similar to the flow prediction module, using both cross and self-attention layers yields better results than using only cross-attention layers. When stacking 4 cross-attention and 2 self-attention layers, we get the best performance. Tab. 4 provides the ablation of different implementation choices, as mentioned in section 4.2. We see that providing supervision of each agent, using a multi-scale strategy, and using a stronger ResNet backbone can improve detection performance. Specifically, we noticed that using single-scale feature fusion has a large performance drop for the AP@0.7 metric.

D. Detailed Comparison of 3D detection performance

To assess the effectiveness of LRCP, we conducted thorough experiments on the V2X-Sim and Dair-V2X datasets. Tab. 6 and 7 delve into the object detection performance under varying latency constraints, ranging from 0 to 500 milliseconds. These tables directly correspond to the tabular data presented in Figure 6 of the main text. Our experiments reveal that LRCP consistently outperforms existing methods on both datasets. This advantage becomes even more significant as the allowable processing time for detection increases.

Table 3. Compensation performance for objects with different speeds on Dair-V2X dataset

Speed	$v \leq 5\,m/s$				$5\,m/s < v \leq 10\,m/s$				$v \geq 10\,m/s$			
Metric	AP@0.5											
Latency (ms)	0	100	300	500	0	100	300	500	0	100	300	500
Deformable attn	79.1	78.8	77.0	71.6	90.5	88.7	51.7	50.3	90.2	71.0	39.8	59.8
LRCP	78.7	78.7	78.5	78.2	90.2	90.3	89.1	86.5	88.3	88.9	86.2	77.4
Metric	AP@0.7											
Latency (ms)	0	100	300	500	0	100	300	500	0	100	300	500
Deformable attn	67.9	67.0	64.6	61.3	81.5	59.6	45.1	45.5	81.4	38.1	35.7	54.0
LRCP	67.6	67.3	66.8	66.4	80.5	79.5	76.3	71.6	79.1	76.4	67.9	58.2

Table 4. Effect of the number of self/cross attention layers in the feature fusion decoder, experimented on Dair-V2X datasets

layers	AP@0.5	AP@0.7
3 cross	80.2	68.8
2 cross+1 self	80.5	69.3
6 cross	80.8	69.3
4 cross+2 self	**80.9**	**70.1**
6 cross + 2 self	79.2	68.6

Table 5. Ablation of implementation options, experimented on Dair-V2X datasets

Supervise single	Multi-scale	ResNet backbone	AP@0.5	AP@0.7
✓	✓	✓	**80.9**	**70.1**
	✓	✓	80.1	68.3
✓		✓	79.3	59.6
✓	✓		80.9	68.7

Table 6. Comparison of 3D detection performance on the V2X-Sim

Latency(ms)	0	200	400	600
Single	76.7			
Metric	AP@0.5			
V2X-ViT	88.8	87.2	84.3	82.9
Openv2v	80.0	79.6	78.8	78.2
Where2comm	86.8	84.6	80.8	79.7
Cobevflow	86.8	85.2	83.3	82.4
DeformableAttn+Syncnet	89.4	87.5	84.7	84.3
Deformable attn	89.4	87.5	83.9	82.3
LRCP (ours)	**89.4**	**89.3**	**89.2**	**88.3**
Metric	AP@0.7			
Single	65.8			
V2X-ViT	79.3	75.6	73.9	73.2
Openv2v	68.1	66.9	66.4	66.5
Where2comm	83.8	77.4	75.0	74.4
Cobevflow	83.8	79.2	78.1	76.2
DeformableAttn+Syncnet	87.5	81.0	80.2	79.6
Deformable Attn	87.5	81.4	78.6	78.1
LRCP (ours)	**87.5**	**86.8**	**85.6**	**83.4**

Table 7. Comparison of 3D detection performance on the Dair-V2X

Latency(ms)	0	100	200	300	400	500
Single	67.4					
Metric	AP@0.5					
V2X-ViT	71.5	70.4	69.1	68.1	67.1	66.5
Openv2v	73.0	72.3	71.3	70.2	69.9	69.6
Where2comm	80.1	78.5	73.7	71.2	69.5	68.7
Cobevflow	80.1	79.0	75.6	74.0	73.3	73.1
DeformableAttn+Syncnet	**80.9**	78.7	74.3	72.8	71.6	70.6
Deformable Attn	**80.9**	79.4	74.6	71.3	69.6	68.3
LRCP (ours)	80.5	**80.5**	**80.3**	**80.0**	**79.6**	79.1
Metric	AP@0.7					
Single	58.7					
Latency (ms)	0	100	200	300	400	500
V2X-ViT	54.7	54.0	53.4	53.1	52.9	52.7
Openv2v	58.0	57.2	56.5	56.3	56.4	56.3
Where2comm	67.3	61.2	58.8	58.3	58.0	57.6
Cobevflow	67.3	62.4	61.1	60.3	59.5	59.0
Def. Attn+Syncnet	**70.1**	64.5	62.1	61.9	61.3	61.1
Deformable Attn	**70.1**	64.1	61.0	60.2	59.6	58.9
LRCP (ours)	69.5	**69.1**	**68.4**	**67.8**	**67.2**	**66.5**