A Dataset

As detailed in Section 4.1, we divide the nuScenes dataset into training, prior, and testing segments. Scenes from *Boston Seaport* and *Singapore Queenstown* comprise the training set, while *Singapore Onenorth* and *Singapore Holland Village* contribute to the prior and testing sets. To maintain an balanced distribution of day and night scenes, a subset of *Singapore Holland Village*, including 32 night scenes, is allocated to the training set. This results in a division of 65 night scenes for training and 34 for testing, with night scenes excluded from prior construction. The distribution between prior and testing sets aims to balance overlap and scene diversity, yielding 19/34 scenes for prior/testing in *Singapore Holland Village* and a 63/120 split in *Singapore Onenorth*. Here we list scene names of prior and testing set for reproduction.

`testing scenes: 0001, 0003, 0005, 0006, 0007, 0008, 0011, 0012, 0014, 0015, 0016, 0018, 0021, 0022, 0023, 0025, 0028, 0029, 0034, 0043, 0045, 0046, 0051, 0054, 0059, 0060, 0061, 0120, 0124, 0126, 0127, 0130, 0131, 0132, 0134, 0135, 0138, 0150, 0151, 0152, 0154, 0155, 0157, 0158, 0159, 0160, 0190, 0191, 0192, 0193, 0194, 0221, 0270, 0271, 0272, 0273, 0274, 0275, 0276, 0277, 0278, 0344, 0345, 0346, 0347, 0355, 0356, 0357, 0358, 0359, 0361, 0362, 0363, 0364, 0365, 0366, 0367, 0368, 0371, 0372, 0373, 0374, 0375, 0376, 0377, 0379, 0380, 0381, 0382, 0383, 0385, 0945, 0949, 0952, 0953, 0956, 0957, 0958, 0959, 0960, 0961, 0963, 0966, 0967, 0968, 0969, 0971, 0975, 0976, 0977, 0978, 0979, 0980, 0981, 0982, 0983, 0984, 0989, 0990, 0991, 1045, 1046, 1047, 1048, 1049, 1050, 1051, 1052, 1053, 1054, 1055, 1056, 1057, 1058, 1059, 1060, 1061, 1062, 1063, 1064, 1065, 1066, 1067, 1068, 1069, 1070, 1071, 1072, 1073, 1074, 1075, 1076, 1077, 1104.`

`prior scenes: 0002, 0004, 0009, 0010, 0013, 0017, 0019, 0020, 0024, 0026, 0027, 0030, 0031, 0032, 0033, 0035, 0036, 0038, 0039, 0041, 0042, 0044, 0047, 0048, 0049, 0050, 0052, 0053, 0055, 0056, 0057, 0058, 0121, 0122, 0123, 0125, 0128, 0129, 0133, 0139, 0149, 0195, 0196, 0268, 0269, 0348, 0349, 0350, 0351, 0352, 0353, 0354, 0360, 0369, 0370, 0378, 0384, 0386, 0399, 0400, 0401, 0402, 0403, 0405, 0406, 0407, 0408, 0410, 0411, 0412, 0413, 0414, 0415, 0416, 0417, 0418, 0419, 0947, 0955, 0962, 0972, 0988.`

B NeRF Implementation

Field Details. Following the methodology of Mip-NeRF 360 [1], our implementation adopts a two-stage proposal network for importance sampling, utilizing 128 and 64 sample points in the first and second stages, respectively. The proposal networks operate across resolutions from 2^4 to 2^{10}, featuring a maximum hash map capacity of 2^{20}. These networks are structured into 8 levels, with each

level encoding a 1-dimensional feature. For the main field, we define resolutions ranging from 2^4 to 2^{14}, maintaining the same maximum hash map capacity of 2^{20} but expanding to 10 levels, each with a 4-dimensional feature. All Multi-Layer Perceptrons (MLPs) in our framework consist of two hidden layers and are configured with a hidden dimension of 64.
Optimization. The comprehensive loss function is represented as:

$$\mathcal{L} = \mathcal{L}_{\text{rgb}} + \lambda_{\text{feat}}\mathcal{L}_{\text{feat}} + \lambda_{\text{sky}}\mathcal{L}_{\text{sky}} + \lambda_{\text{inter}}\mathcal{L}_{\text{inter}} + \lambda_{\text{dist}}\mathcal{L}_{\text{dist}} \tag{1}$$

We set the weights for λ_{feat}, λ_{sky}, λ_{inter}, and λ_{dist} at 0.5, 0.001, 1.0, and 0.002, respectively. For optimization, we employ the AdamW optimizer [6] with an initial learning rate of 0.01, applying a decay factor of 0.33 at the 25%, 50%, and 75% marks of the total iterations.

C Baseline Implementation

This section outlines the implementation details for the baseline perception models, closely following their original codebases.
HD Map Construction. Following MapTR [5] and StreamMapNet [8], we adjust image resolution to 450×800 and use ResNet50 [2] as the backbone. We use 24 training epochs, with learning rates set at 5×10^{-4} for MapTR and 6×10^{-4} for StreamMapNet, both with a batch size of 32, following their standard implementation. For the BEVFormer-based rasterized model, we use a batch size of 8 and a learning rate of 2×10^{-4}, consistent with Neural Map Prior [7]. Historical frames are not used for simplicity.
Occupancy Prediction. Following FB-OCC [4], we resize images to 256×704 and use ResNet50 as the image backbone. The models are trained for 24 epochs with BEV data augmentation. The learning rates are 2×10^{-4} for a batch size of 64 for FB-OCC and 1×10^{-4} for a batch size of 32 for BEVDet [3], in line with their official setups. Historical frames are incorporated to align with the original implementations.

D Other Ablations

D.1 Localization Errors

In this section, we discuss the impact of localization errors on our method, focusing on two types:
Pose errors in prior data before optimizing NeRFs. When building NeRFs offline, techniques like visual SLAM and visual odometry can reduce localization errors to under 10 cm, which our NeRFs can handle. For example, the NuScenes dataset lacks height information in ego-poses, resulting in about a 10 cm error in camera poses, yet our NeRFs manage this well.
Localization noise during online perception integration. Autonomous vehicles using GNSS with IMU or RTK typically have pose errors of 0.3 m to

Table 1: Performance for online models with noisy localization.

Model	mAP/mIoU		
	w.o. Noise	w. Noise	w.o. Priors
StreamMapNet	25.7	24.5	11.1
BEVDet	33.7	32.5	29.3

Table 2: Experiments on priors-only models. "Setting a" evaluates models trained with both online observations and priors, replacing the online BEV features with zeros. "Setting b" involves training new models using only priors during both training and testing. "Full" represents our best models. "Baseline" denotes models without any priors.

Model	mAP			
	Setting a	Setting b	Full	Baseline
StreamMapNet	1.9	19.8	25.7	11.1
MapTR	6.0	18.8	23.1	7.7

0.5 m. To test our models' robustness to localization noise, we added 2D Gaussian noise of $N(0, 0.5^2)$ to the ego-poses during evaluation. As shown in Table 1, while localization noise slightly impacts performance, models with priors still outperform those without.

D.2 Performance without Online Observation

This section examines the performance of models using only priors, without online observations. We consider two settings: (a) Evaluating models trained with both online observations and priors, but replacing the online BEV features with zeros. (b) Training new models using only priors, ignoring the "encoding online observation" part during both training and testing, serving as "look-up" models or automatic map annotators. Results in Table 2 show that models using only priors at test time perform poorly. However, models trained with priors only perform decently, indicating the potential for using priors to automatically annotate maps. Note that these results are evaluated where X_{prior} and X_{test} overlap, and online features are still necessary where priors do not cover.

References

1. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. CVPR (2022) 1
2. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016) 2

3. Huang, J., Huang, G.: Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. arXiv preprint arXiv:2203.17054 (2022) 2
4. Li, Z., Yu, Z., Austin, D., Fang, M., Lan, S., Kautz, J., Alvarez, J.M.: FB-OCC: 3D occupancy prediction based on forward-backward view transformation. arXiv:2307.01492 (2023) 2
5. Liao, B., Chen, S., Wang, X., Cheng, T., Zhang, Q., Liu, W., Huang, C.: Maptr: Structured modeling and learning for online vectorized hd map construction. In: International Conference on Learning Representations (2023) 2
6. Loshchilov, I., Hutter, F.: Fixing weight decay regularization in adam (2018) 2
7. Xiong, X., Liu, Y., Yuan, T., Wang, Y., Wang, Y., Hang, Z.: Neural map prior for autonomous driving. In: CVPR (2023) 2
8. Yuan, T., Liu, Y., Wang, Y., Wang, Y., Zhao, H.: Streammapnet: Streaming mapping network for vectorized online hd map construction. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 7356–7365 (January 2024) 2