Learning 1-Bit LiDAR-based Localization with Auxiliary Objective

Kaijie Yin[1,2,3], Zhiyuan Zhang[2], Tian Gao[1], Wentao Zhu[3], Cheng-zhong Xu[1], and Hui Kong[1*]

[1] University of Macau
[2] Singapore Management University
[3] Eastern Institute of Technology, Ningbo

Abstract. 6-DoF LiDAR-based localization is a fundamental capability for autonomous systems operating in large-scale outdoor environments. Many deep-learning-based localization methods have achieved promising performance so far. However, as one of the always-on modules competing for limited on-board computational resources, the localization module is expected to consume only a small portion of the overall compute budget. Most existing learning-based methods are still too heavy for this purpose. In contrast, binary neural networks (BNNs) offer an appealing solution, but the 1-bit compression causes severe information loss and performance drop. In this paper, we address this challenge by proposing Binarized LiDAR-based Localization (BiLoc), the first binary neural network framework for 6-DoF LiDAR localization. Specifically, we reinterpret the training of BNNs from the perspective of the information-bottleneck principle, aiming at retaining minimal yet sufficient representations for pose estimation while suppressing redundant variations. And we introduce an auxiliary objective that adaptively regulates information retention in the binary encoder, effectively mitigating the information loss caused by binarization. This auxiliary objective provides additional optimization signals that compensate for the limited representational capacity and the gradient mismatch inherent in BNNs. Extensive experiments on large-scale outdoor LiDAR datasets demonstrate that BiLoc establishes a new state of the art for LiDAR localization with BNNs.

1 Introduction

Modern autonomous driving systems [53, 59] are typically designed in a modular paradigm where perception, localization, planning, and control are implemented as loosely coupled components. Among these modules, localization plays a safety-critical and always-on role, as accurate pose estimation is required continuously for downstream decision making. Unlike perception backbones that can be selectively activated or offloaded, the localization module must operate under strict latency, power, and memory constraints. These constraints make lightweight and efficient localization a fundamental requirement in autonomous systems.

* Corresponding author.

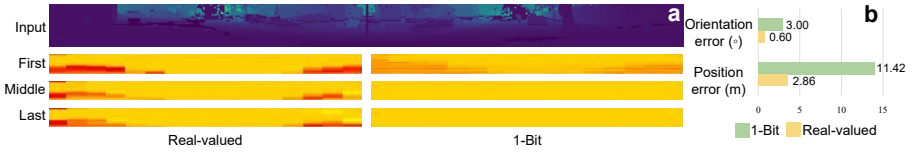

Fig. 1: (a) Visualization of the information loss of real-valued ViT-S/16 and ReActNet-based ViT-S/16 on the Oxford Radar RobotCar dataset [1] at the first, middle, and last blocks [12]. Darker colors indicate less discarded information, showing higher importance for LiDAR localization. (b) Comparison of localization accuracy of full-precision ViT-S/16 and ReActNet-based ViT-S/16 on the 18-14-14-42 test split of the Oxford Radar RobotCar dataset.

Traditional LiDAR localization methods [68, 73] rely on explicit map construction and point-to-map registration, incurring high computational cost and frequent map maintenance. With the rise of deep neural networks, learning-based approaches have achieved significant progress. These methods can be broadly categorized into Scene Coordinate Regression (SCR) [5,6,72] and Absolute Pose Regression (APR) [36,63,64]. SCR predicts point-to-scene correspondences followed by a 6-DoF pose estimation via classical registration [19]. In contrast, APR directly regresses the global pose in an end-to-end manner without explicit map matching. However, existing APR models remain computationally expensive for deployment on resource-constrained edge devices, motivating more aggressive model compression strategies.

Existing compression techniques mainly include pruning [24, 26], knowledge distillation [27,51], quantization [23,29,39], and compact network design [42,47]. Model quantization is a promising solution, which significantly reduces memory usage and computational cost by quantizing weights and/or activations into low-bit integers. The most extreme form of quantization is binarization [28, 41, 45], which encodes both weights and activations using 1-bit representations. BNNs replace arithmetic operations with efficient logic operations (XNOR and PopCount), achieving up to 32× memory compression and 58× computational savings [46]. These advantages make binarization attractive for localization modules that face limited-resource challenges. However, directly applying existing binarization techniques to LiDAR-based localization leads to severe performance degradation. The primary limitation lies in the insufficient expressiveness of binary feature representations, which is particularly problematic for localization as a regression task requiring fine-grained information to resolve subtle pose differences. In practice, this limitation is further aggravated by optimization difficulties during BNN training. Aggressive binarization restricts the representation space, while the non-differentiable sign function causes gradient mismatch. As a result, standard training objectives struggle to learn task-sufficient representations, leading to unstable optimization and poor localization accuracy.

To investigate this issue, we analyze BNNs for LiDAR localization from an information-theoretic perspective. We adopt an information-bottleneck based analysis method [56], Quantifying Knowledge Points [80], to compare a real-

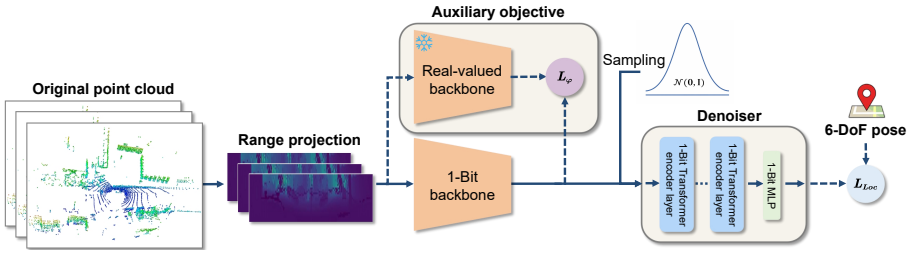

Fig. 2: BiLoc framework: The raw LiDAR point cloud is first projected into a range image that preserves geometric and depth information. The range image is fed into a binarized backbone to extract global features. These features are then passed to a fully binarized PoseDiffusion [62] decoder to predict the global pose. During training, the backbone features are also supervised by an auxiliary objective, which provides additional optimization guidance without introducing inference overhead.

valued ViT-S/16 model [17] with its binarized counterpart, a ReActNet-based ViT-S/16 model [33, 40]. As shown in Fig. 1, the binarized model exhibits progressive information loss across layers during forward propagation. Compared with the real-valued network, the 1-bit model shows weaker ability to capture task-relevant information even in shallow blocks. This issue becomes more severe in deeper layers, where limited representation capacity and gradient mismatch further amplify information loss. These observations suggest that the main challenge of 1-bit localization lies not only in reduced capacity, but also in the lack of mechanisms to regulate information retention under extreme compression.

Based on these observations, we propose Binarized LiDAR-based Localization (BiLoc). Instead of training BNNs with a single objective, BiLoc formulates the learning process under the information-bottleneck principle [52, 55], aiming to retain minimal yet task-sufficient representations for pose estimation while suppressing irrelevant variations. To facilitate this objective, we introduce an auxiliary objective that performs feature distillation using representations from an offline real-valued model as a reference distribution. This additional supervision compensates for the limited representational capacity and gradient mismatch of BNNs, guiding the binarized encoder to focus on components with larger information loss. Importantly, the auxiliary objective is used only during training and removed after convergence, introducing no extra cost during inference. Consequently, BiLoc preserves the efficiency advantages of 1-bit networks while significantly improving representation quality. As illustrated in Fig. 2, extensive experiments on large-scale outdoor LiDAR datasets demonstrate that BiLoc establishes a new benchmark for BNN-based LiDAR localization. The main contributions are summarized as follows:

- We propose the BiLoc method, the first BNN framework tailored for outdoor localization. By constraining both weights and activations to 1-bit, BiLoc substantially reduces computational and memory costs while maintaining competitive localization accuracy.

- We identify that the primary challenge of applying BNNs to LiDAR localization lies in their limited representational capacity and optimization difficulty. To address this, we introduce a training-only auxiliary objective that promotes the preservation of task-relevant information at the representation level without adding inference overhead.
- We conduct extensive experiments on large-scale outdoor LiDAR datasets to validate the effectiveness of the proposed framework. BiLoc consistently outperforms existing binarization-based localization methods and establishes a new state-of-the-art (SOTA) in this field. In particular, on the Oxford Radar RobotCar dataset, BiLoc reduces the averaged mean position error by 10.11% and the averaged mean orientation error by 11.16% compared with the strongest binarized baseline.

2 Related Work

Learning-based localization can be broadly divided into structure-based Scene Coordinate Regression (SCR) and regression-based Absolute Pose Regression (APR). SCR estimates the global pose by performing 3D matching between LiDAR point clouds and the world coordinate system. Most mainstream methods [11, 32, 58, 65] rely on learned descriptors and require storing point cloud descriptors for pose estimation. SGLoc [37] is the first method to regress correspondences with a neural network and estimate pose with RANSAC, inspiring follow-up works [35,71,72]. However, since only the correspondence stage is trainable, its accuracy is significantly affected by the RANSAC module. In contrast, APR regresses the 6-DoF pose end-to-end [9, 30, 31, 43, 49, 61], avoiding point cloud registration and thus being more efficient at inference. Since LiDAR is robust to illumination changes, LiDAR-based APR methods have achieved strong performance in large-scale outdoor environments. PointLoc first introduces APR for LiDAR localization, while [78] explores memory-efficient regression schemes with four models. HypLiLoc fuses multi-modal scene features in hyperbolic and Euclidean spaces. DiffLoc further adopts vision foundation models [44] for feature extraction and uses diffusion models [62] for iterative pose regression.

Auxiliary objective was initially introduced [54] to address the difficulty of training deep neural networks. These objectives are only active during training and are removed after training, so they do not introduce extra computation during inference. In recent years, auxiliary objectives have been widely used in local learning [2,18,66], where gradient-isolated submodules are trained with module-specific losses to avoid cross-module backpropagation. Beyond local learning, auxiliary objectives have also been linked to knowledge distillation (KD). Zhang et al. [79] employ multi-exit self-distillation, where intermediate exits act as auxiliary supervision, and Deng et al. [16] further formalize KD as a training-time auxiliary objective to provide more reliable gradients for spiking neural networks [16]. In BNNs, auxiliary designs have been explored for object detection [70] and change detection [75], demonstrating their effectiveness in improving optimization and performance.

Binary neural networks (BNNs) represent the extreme of quantization that reduces both weights and activations to 1-bit, substantially lowering computation and memory but often incurring notable accuracy drops compared with real-valued models. To mitigate this problem, many approaches have been proposed. For instance, scaling factors are applied to weights and activations to alleviate quantization error [46]; real-valued identity shortcuts are introduced to enhance representation power [41]; and learnable thresholds together with RPReLU activations are adopted to better handle different data distributions [40]. For Vision Transformers (ViTs), directly applying binarization often leads to severe performance degradation [21], motivating a series of dedicated designs [25, 38, 76]. Despite this progress, most methods are validated primarily on classification, while their effectiveness on downstream tasks [10, 20, 69, 74], especially localization, remains underexplored. To bridge this gap, we present a training optimization framework with an auxiliary objective tailored for 1-bit LiDAR localization networks, improving the performance without any additional inference cost.

3 Preliminaries

3.1 Binary Neural Networks

For the i-th layer, let W_i and A_i be the real-valued weights and activations. Binary neural networks quantize both to 1-bit, replacing floating-point multiplications with efficient bitwise operations (XNOR and PopCount).

As the fundamental module in 1-bit ViTs, the binary linear layer ($\Lambda(\cdot)$) can be expressed as $A_{i+1}^b = \Lambda(A_i, W_i) = \varsigma \odot \left(A_i^b \circledast W_i^b\right)$, where A_i^b and W_i^b are binarized activations and weights, ς is a channel-wise scaling factor, $\circledast$ denotes the bitwise operation based on XNOR and PopCount, as shown in Fig. 3. $\odot$ is element-wise multiplication. To obtain binarized activations A^b, we apply the sign function in the forward pass and use a piecewise polynomial function in the backward pass, as follows:

$$\text{Forward}: \mathrm{A^b} = sign\left(\frac{\mathrm{A}-\beta}{\alpha}\right),$$
$$\text{Backward}: \frac{\partial L}{\partial \mathrm{A}} = \frac{\partial L}{\partial \mathrm{A^b}} \odot \frac{\partial \mathrm{A^b}}{\partial \mathrm{A}} = \begin{cases} \frac{\partial L}{\partial \mathrm{A^b}} \odot \left(2 + 2\left(\frac{\mathrm{A}-\beta}{\alpha}\right)\right), & \beta - \alpha \leqslant \mathrm{A} < \beta \\ \frac{\partial L}{\partial \mathrm{A^b}} \odot \left(2 - 2\left(\frac{\mathrm{A}-\beta}{\alpha}\right)\right), & \beta \leqslant \mathrm{A} < \beta + \alpha \\ 0, & \text{otherwise} \end{cases}, \tag{1}$$

where α and β are learnable scaling and bias terms, respectively. Here L is the loss function. The binarized weight W^b commonly computed as follows:

$$\text{Forward}: \mathrm{W^b} = \rho\left(abs\left(\mathrm{W}\right)\right) \odot sign\left(\mathrm{W}\right), \text{Backward}: \frac{\partial L}{\partial \mathrm{W}} = \rho\left(abs\left(\mathrm{W}\right)\right) \odot \frac{\partial L}{\partial \mathrm{W^b}} \odot \mu, \tag{2}$$

where $\rho(\cdot)$ denotes the average function used to compute the per-channel scaling factor of the weights. μ is a mask tensor with the same shape as W. An entry in the mask is set to one if the corresponding value in W lies within the interval $[-1, 1]$.

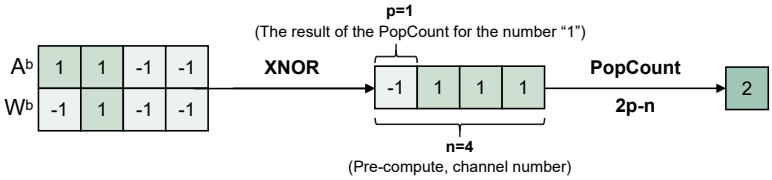

Fig. 3: The inner product of two binary vectors can be efficiently computed using XNOR and PopCount. After the XNOR operation, let p be the number of matched bits (PopCount result) and u be the number of unmatched bits, where $n = p + u$ is the vector length. Since $u = n - p$, the dot product can be expressed as $p - (n - p)$.

3.2 Information-bottleneck Theory

Information-bottleneck (IB) theory provides a unified framework for balancing data compression and task-relevant information preservation. The mutual information (MI) is a measure of the shared information between two random variables X and Y, defined as (for discrete variables)

$$I(X,Y) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)\,p(y)} = \sum_x p(x) \sum_y p(y|x) \ \log \left(\frac{p(y|x)}{p(y)} \right). \quad (3)$$

The IB theory can be viewed as a special case of rate-distortion (RD) theory [4, 22], which trades off lossy compression and distortion. RD seeks a compact representation T of a source X via $p(t|x)$ by minimizing $I(X,T)$ subject to an expected distortion constraint $E[d(X,T)] \leq \epsilon^*$. Here, $d(x,t)$ measures the deviation between X and T. A key limitation of RD is that both the distortion function $d(x,t)$ and the threshold ϵ^* must be specified a priori, which can be difficult without prior knowledge [55]. IB addresses this by introducing task supervision Y and a trade-off parameter σ, compressing X into T while preserving task-relevant information, formulated as

$$\mathbf{IB}(p(t|x)) = \min I(X,T) - \sigma I(T,Y), \quad (4)$$

where $I(X,T)$ measures the compression of X into T, and $I(T,Y)$ quantifies the retention of task-relevant information. The parameter σ controls the trade-off, and $p(t|x)$ defines the mapping from x to t. Previous studies [52, 80] have shown that deep networks can learn efficient representations related to minimal sufficient statistics from the IB perspective.

4 Methodology

4.1 Problem Formulation

Tishby et al. [52] suggest that the hierarchy of a DNN can be viewed as a (continuous) Markov chain that progressively refines data representations. In LiDAR localization, intermediate encoder blocks produce latent features $Z = f(X)$ from

an input point cloud frame X, retaining task-relevant information while discarding redundancy. As illustrated in Fig. 1a, regions critical to performance, where perturbations cause noticeable degradation, tend to shrink with increasing depth. A regressor $g(Z)$ then predicts the global pose Y, yielding the mapping $X \xrightarrow{f(X)} Z \xrightarrow{g(Z)} Y$. Under the IB principle, Z is encouraged to be a minimal sufficient statistic of Y. Following Eq. 4, this corresponds to maximizing $I(Z, Y)$ while minimizing $I(X, Z)$, which can be formulated by

$$\mathbf{IB}(X, Y, Z) = \min_{\theta_f, \theta_g} I(X, Z) - \sigma I(Z, Y), \tag{5}$$

where θ_f and θ_g are the parameters of the encoder and regressor, respectively, and $I(\cdot, \cdot)$ denotes mutual information. $I(X, Z)$ encourages compressing X into Z by suppressing task-irrelevant redundancy, while $I(Z, Y)$ promotes preserving task-relevant information for predicting Y. In LiDAR localization, maximizing $I(Z, Y)$ can be instantiated by the standard localization loss $L_{\text{localization}}$.

However, Fig. 1b shows that BNNs achieve much lower localization accuracy than real-valued networks. Excessive information discard yields a very low $I(X, Z)$, leaving little room for further optimization; meanwhile, important task-related information is poorly preserved, making $I(Z, Y)$ hard to maximize. This effect is partly illustrated on the right part of Fig. 1a. This gap stems from the binarization mechanism in Eq. 1 and Eq. 2: the forward pass relies on the sign function for both weights and activations, limiting representational capacity. In the backward pass, the sign function is approximated by the straight-through estimator (STE) [3] and variants, whose gradients deviate from true gradients and introduce cumulative errors. This causes the intrinsic gradient mismatch problem of BNNs, leading to unstable training and degraded performance.

4.2 Auxiliary Objective

Auxiliary objective Design. To alleviate the above issues, we introduce a feature-distillation-based auxiliary objective for training BNNs in LiDAR localization. Specifically, we train a real-valued teacher with a strong vision foundation encoder [44] on the localization task to obtain a reference model. We then use its features as a reference distribution to guide the training of the binarized encoder. Following [50], we adopt the efficient offline distillation scheme as the basis of the auxiliary objective. We view the real-valued features as an approximation to the optimal sufficient statistic, providing an approximate upper bound for $I(X, Z)$ in Eq. 5. Based on this, BNN training can be reformulated as:

$$\min_{\theta_f, \theta_g} I(X, Z) - \sigma_1 I(Z, Y) - \sigma_2 \varphi, \tag{6}$$

where σ_1 and σ_2 are trade-off coefficients balancing information compression and task-relevant information retention. We define a structural operator $S(\cdot)$ to capture the inter-sample relational structure of latent features, which are further used to construct a cost matrix C. Here, φ denotes the auxiliary objective:

$$\varphi = I(Z_{diff}, Z^*) + I(S(Z), S(Z^*) \mid \mathbf{p}), \tag{7}$$

where Z_{diff} denotes the importance-weighted latent feature by a soft-mask. Z^* denotes the latent feature of the offline real-valued model, which is treated as an approximation of the optimal sufficient statistic. $I(Z_{\text{diff}}, Z^*)$ is a soft-masked feature distillation term for the BNN encoder to learn the latent features of offline real-valued counterparts, which emphasizes important regions while suppressing variations in irrelevant regions. $I(S(Z), S(Z^*) \mid \mathbf{p})$ is a task-specific conditional mutual information term. With pose as side information, it relaxes the strict one-to-one correspondence between samples in Z and Z^*, denoted as z and z^*, encouraging each sample in Z to align not only with its corresponding sample in Z^* but also with its neighboring samples, which have similar poses.

Auxiliary Objective Loss. Combining Eq. 6 and Eq. 7, we obtain the final objective for BiLoc training. We instantiate the auxiliary objective in Eq. 7 with a feature distillation loss using a soft mask and a structure distillation loss. For $I(Z_{\text{diff}}, Z^*)$, the auxiliary objective encourages the BNN encoder to learn task-critical components from the offline teacher features while suppressing redundant information. In BiLoc, these components correspond to different feature channels. Since obtaining a knowledge-points mask [80] is extremely time-consuming during training, we instead compute the channel-wise Mahalanobis distance between real-valued features and their 1-bit counterparts, and apply a sigmoid function to obtain a soft mask:

$$\mathbf{m} = \varsigma\big(d_M^2(Z_c^*,\, Z_c;\, \mathbf{\Sigma}_c)\big)\,, \quad d_M^2(\mathbf{a}, \mathbf{b}; \mathbf{\Sigma}) = (\mathbf{a}-\mathbf{b})^\top \mathbf{\Sigma}^{-1}(\mathbf{a}-\mathbf{b}), \tag{8}$$

where $\mathbf{m}$ denotes the mask, c denotes the channel, $\varsigma(\cdot)$ denotes the sigmoid function, and $d_M^2(\cdots)$ denotes the Mahalanobis distance. In practice, all channels share a learnable variance scalar for computational efficiency. We apply this mask to reweight features for distillation and use an L_1 loss as the distillation objective, which approximately minimizes $-I(Z_{\text{diff}}, Z^*)$: $||\mathbf{m} \odot Z - Z^*||_1$.

For $I(S(Z), S(Z^*) \mid \mathbf{p})$, we instantiate this term based on the optimal transport principle [60]. Specifically, following prior work [8], we model the structural alignment between teacher and student features as a primal optimal transport problem. It can be interpreted as finding an optimal matching that transfers probability mass from $S(Z)$ to $S(Z^*)$ when only limited training samples are available. In practice, we follow previous studies [48] to use the Sinkhorn algorithm [15] to obtain an efficient approximation. The Sinkhorn algorithm performs iterative normalization over rows and columns, which is equivalent to applying softmax operations along both dimensions. We denote the Sinkhorn operator as $\|\cdot, ; C\|_s$, where $C \in \mathbb{R}^{N \times M}$ is the cost matrix that measures the discrepancy between two feature sets. Specifically, the matching cost between the i-th teacher sample Z_i^* and the j-th student sample Z_j is defined as

$$C_{ij} = \big(1 - \cos(Z_i^*, Z_j)\big) + d_{\text{pose}}(Z_i^*, Z_j), \tag{9}$$

where i and j denote the sample indices in the teacher and student batches, respectively. Here, the first term is the cosine cost that measures the feature discrepancy between Z_i^* and Z_j. The second term is a pose-aware cost tailored

for LiDAR localization, which encodes the geometric inconsistency between the corresponding poses, which is defined as:

$$d_{\text{pose}}(Z_i^*, Z_j) = \|t_i - t_j\|_2 + \theta'(q_i, q_j), \quad \theta'(q_i, q_j) = 2\arccos\big(|q_i^\top q_j|\big), \tag{10}$$

where t is the ground-truth translation vector of each sample, and q is the corresponding rotation represented as a quaternion. $\theta'(\cdot)$ denotes the minimal rotation angle between two poses. Under this setting, the loss function can be described as:

$$||S(Z), S(Z^*); C||_s \triangleq \underset{\pi \in \mathbf{\Pi}(S(Z), S(Z^*))}{\arg\min} \langle \pi, C \rangle + \varepsilon \sum_{i,j} \pi_{ij} \log(\pi_{ij}), \tag{11}$$

where π denotes the discrete joint distribution of $S(Z)$ and $S(Z^*)$, which is also known as the transport plan. It satisfies the constraint: $\mathbf{\Pi}(S(Z), S(Z^*)) = \big\{\pi \mid \pi\mathbf{1} = S(Z), \pi^\top \mathbf{1} = S(Z^*)\big\}$, where $\mathbf{1}$ denotes an all-ones vector such that the matrix–vector multiplication enforces the marginal (row and column sum) constraints. The term $\langle \pi, C \rangle = \sum_{i,j} \pi_{i,j} C_{i,j}$ denotes the total matching cost, which is computed in practice using the Frobenius dot product. The last term is a convex regularization term in the Sinkhorn algorithm. We use the optimal π as a proxy loss to approximately minimize $-I(S(Z), S(Z^*) \mid \mathbf{p})$.

To clarify the formula's meaning, we repositioned the hyperparameters in the following equation. Accordingly, we optimize the 1-bit LiDAR localization mode based on the total loss as:

$$\mathcal{L}_{total} = \underbrace{\mathcal{L}_{\text{localization}}}_{I(X;Z) - \sigma I(Z;Y)} + \lambda_1 \underbrace{\|\mathbf{m} \odot Z - Z^*\|_1}_{-I(Z_{\text{diff}}; Z^*)} + \lambda_2 \underbrace{\|S(Z), S(Z^*); C\|_s}_{-I(S(Z), S(Z^*) | \mathbf{p})}. \tag{12}$$

Theoretical Analysis. We consider a binarized network decomposed into an encoder $f_e(\cdot; \theta_e)$ and a task head $f_c(\cdot; \theta_c)$. Given an input X and label y, the prediction is $y_c = f_c(f_e(X; \theta_e); \theta_c)$. Due to the non-differentiability of the sign function, training relies on the Straight-Through Estimator (STE), which introduces systematic gradient mismatch during backpropagation through binarized layers, as shown in Eq. 1 and 2. Let G denote an "accurate" reference gradient for the encoder, and let υ represent the accumulated STE-induced error. The encoder-side gradient propagated from the task head can be modeled as $\text{STE}\Big(\frac{\partial y_c}{\partial \theta_e}\Big) = G + \upsilon$. For baseline training using only the task loss $\mathcal{L}_{\text{localization}}$, which denoted as $\mathcal{L}_{\text{loc}}$ for simplicity, the encoder receives $\frac{\partial \mathcal{L}_{\text{loc}}}{\partial \theta_e} = \frac{\partial \mathcal{L}_{\text{loc}}}{\partial y_c}(G + \upsilon)$, and the relative gradient mismatch can be characterized by $K_{\text{base}} = \|\upsilon\| / \|G\|$.

As mentioned in earlier chapters, BiLoc introduces auxiliary objectives defined directly on the encoder representation to provide additional gradient routes independent of the task head, forming the total objective shown in Eq. 12. Let

$$\Phi_{\text{rep}} = \frac{\partial Z}{\partial \theta_e} \frac{\partial \mathcal{L}_{\text{rep}}}{\partial Z}, \quad \Phi_{\text{struct}} = \frac{\partial Z}{\partial \theta_e} \frac{\partial \mathcal{L}_{\text{struct}}}{\partial Z} \tag{13}$$

denote the encoder gradients contributed by representation distillation and structure-aware supervision, respectively, where $\mathcal{L}_{\text{rep}}$ represents $-I(Z_{\text{diff}}; Z^*)$, and

$\mathcal{L}_{\text{struct}}$ represents $-I(S(Z), S(Z^*) \mid \mathbf{p})$. These gradients bypass the task head and avoid accumulating head-side STE mismatch, providing more stable auxiliary signals for updating θ_e. The overall encoder gradient becomes

$$\frac{\partial \mathcal{L}_{\text{total}}}{\partial \theta_e} = \frac{\partial \mathcal{L}_{\text{loc}}}{\partial y_c}(G + \upsilon) + \lambda_1 \Phi_{\text{rep}} + \lambda_2 \Phi_{\text{struct}}. \tag{14}$$

Accordingly, we define the relative mismatch proxy as

$$K_{\text{BiLoc}} = \| \upsilon \| \, / \, \| G + \lambda_1 \Phi_{\text{rep}} + \lambda_2 \Phi_{\text{struct}} \| . \tag{15}$$

By the vector squared sum identity $\|a+b\|^2 = \|a\|^2 + \|b\|^2 + 2\langle a, b\rangle$, we have

$$\|G + \lambda_1 \Phi_{\text{rep}} + \lambda_2 \Phi_{\text{struct}}\|^2 = \|G\|^2 + \|\lambda_1 \Phi_{\text{rep}} + \lambda_2 \Phi_{\text{struct}}\|^2 + 2\langle G, \lambda_1 \Phi_{\text{rep}} + \lambda_2 \Phi_{\text{struct}}\rangle. \tag{16}$$

Therefore, a sufficient condition for $\|G + \lambda_1 \Phi_{\text{rep}} + \lambda_2 \Phi_{\text{struct}}\| \geq \|G\|$ is

$$\left\langle G, \lambda_1 \Phi_{\text{rep}} + \lambda_2 \Phi_{\text{struct}} \right\rangle \geq -\frac{1}{2}\|\lambda_1 \Phi_{\text{rep}} + \lambda_2 \Phi_{\text{struct}}\|^2. \tag{17}$$

In particular, when the auxiliary gradients are not adversarial to the task gradient in the sense that $\left\langle G, \lambda_1 \Phi_{\text{rep}} + \lambda_2 \Phi_{\text{struct}} \right\rangle \geq 0$, the inequality holds directly, yielding $K_{\text{BiLoc}} \leq K_{\text{base}}$. In practice, representation-level supervision enlarges the effective gradient magnitude, while the structure-aware components reduce misaligned or noisy auxiliary gradients, making them more consistent with the task gradient. As a result, BiLoc alleviates the relative impact of STE-induced gradient mismatch in deep binarized encoders by injecting stable and task-relevant auxiliary gradients at the representation level.

5 Experiments

5.1 Experimental Setup

Dataset and Metrics. We evaluate proposed BiLoc for LiDAR localization on two large-scale outdoor datasets: Oxford Radar RobotCar [1], which we denote as Oxford for simplicity, and NCLT [7] datasets. During the evaluation, we select the mean position and orientation error as the evaluation metrics.

Oxford Radar RobotCar dataset is a widely used benchmark for urban localization. It covers about 2 km^2, and each trajectory is nearly 10 km long. The dataset provides multi-sensor data, including LiDAR, cameras, radar, and GPS/INS. In our work, we use only LiDAR data. Its diverse traffic conditions make it suitable for evaluating LiDAR-based localization methods. Following prior studies [37, 72], we use sequences 11-14-02-26, 14-12-05-52, 14-14-48-55, and 18-15-20-12 for training, and 15-13-06-37, 17-13-26-39, 17-14-03-00, and 18-14-14-42 for testing.

NCLT dataset is a campus area localization dataset collected at the Michigan North Campus. It consists of data from the LiDAR, omnidirectional camera, and

GPS/INS. In our experiments, we use only LiDAR information. It includes 27 tracks, and each track is about 5.5 km long, covering an area of 0.45 km^2. The dataset contains many challenging scenes, such as moving objects and seasonal variations, and structural changes caused by construction.

Implementation details. Our method is implemented in PyTorch. The range image size and patch size are set to $[32, 512]$ and $[4, 16]$, respectively, with a batch size of 50. The input point cloud sequence consists of three frames with a stride of two. The number of denoising steps is set to 10 on Oxford and 15 on NCLT. We train the model for 200 epochs using AdamW [17] with an initial learning rate of 1e-3 and a cosine annealing schedule on a single RTX 5090 GPU. For the auxiliary objective, the offline teacher encoder is DINO [44] with a ViT-S/16 backbone, pre-trained on the corresponding dataset, while the rest of the network is trained from scratch. The auxiliary objective follows the same initial learning rate and warm-up schedule as the backbone, and its gradient update to the backbone stops after the 80th epoch [14].

5.2 Comparison with SOTA BNNs

Baselines and comparisons. As the first study to explore LiDAR localization based on BNNs, we compare our proposed BiLoc with existing ViT-adapted BNNs, including ReActNet [40], BinaryViT [33], and BHViT [21], which have achieved state-of-the-art performance on classification tasks. We adapt ReActNet to the ViT-S/16 [57] architecture for our experiments. We re-run the official code released by the authors of these SOTA BNNs and train all models under the same settings. We adopt DiffLoc [36], an SOTA APR method, as the evaluation framework and integrate BiLoc and all baseline BNNs into this framework for fair comparison. The results of real-valued networks are also included as references. Memory usage follows the mainstream approach [40,67]: 32× real-valued kernels + 1× binary kernels. Operations (OPs) are calculated as real-valued FLOPs plus 1/64 of 1-bit multiplications, following Bi-Real-Net's protocol [41].

Table 1 compares the computational complexity and memory consumption of different quantization strategies and LiDAR localization frameworks. Benefiting from a lighter backbone and 1-bit quantization, our method achieves significant improvements in both computational efficiency and memory usage compared with its real-valued version. This improvement effectively alleviates the computational bottleneck of real-valued DiffLoc and makes the framework more suitable for deployment in resource-constrained environments.

Results on the Oxford dataset. Table 2 evaluates the performance of the proposed method on the Oxford dataset. We report the mean position error and orientation error averaged over all test trajectories. Our method achieves an averaged mean error of 7.56m/1.91°, reducing the error of the previous SOTA BNNs method by 0.85m/0.24°, without introducing any additional computational or memory overhead.

Results on the NCLT dataset. Since the NCLT dataset covers various seasonal changes and spans a longer time period, it poses a greater challenge for localization compared with the Oxford dataset. As shown in Table 3, our method

Table 1: Comparison of inference-time model complexity for real-valued and 1-bit LiDAR localization models. Params are reported in millions (M), and OPs are reported in giga-operations (G).

Real-valued Models			1-bit Models		
Method	Params (M)	OPs (G)	Method	Params (M)	OPs (G)
PointLoc	3.29	10.13	DiffLoc + ReActNet	2.89	0.38
STCLoc	9.31	1.59	DiffLoc + BinaryViT	2.83	0.14
HypLiLoc	52.31	4.89	DiffLoc + BHViT	3.01	0.17
DiffLoc	39.96	77.06	DiffLoc + Ours	3.01	0.17

Table 2: Quantitative results of real-valued LiDAR localization models, 1-bit LiDAR localization models, and our proposed methods. The evaluation covers the mean position(m) and orientation(°) error on the Oxford dataset.

Framework	Method	Bits	15-13-06-37	17-13-26-39	17-14-03-00	18-14-14-42	Average [m/°]
PointLoc [64]	Real-valued	32	12.42/2.26	13.14/2.50	12.91/1.92	11.31/1.98	12.45/2.17
PosePN [78]	Real-valued	32	14.32/3.06	16.97/2.49	13.48/2.60	9.14/1.78	13.48/2.48
PosePN++ [78]	Real-valued	32	9.59/1.92	10.66/1.92	9.01/1.51	8.44/1.71	9.43/1.77
PoseMinkLoc [78]	Real-valued	32	11.20/2.62	14.24/2.42	12.35/2.46	10.06/2.15	11.96/2.41
PoseSOE [78]	Real-valued	32	7.59/1.94	10.39/2.08	9.21/2.12	7.27/1.87	8.62/2.00
STCLoc [77]	Real-valued	32	6.93/1.48	7.55/1.23	7.44/1.24	6.13/1.15	7.01/1.28
HypLiLoc [63]	Real-valued	32	6.88/1.09	6.79/1.29	5.82/0.97	3.45/0.84	5.74/1.05
DiffLoc [36]	Real-valued	32	3.57/0.88	3.65/0.68	4.03/0.70	2.86/0.60	3.53/0.72
	ReActNet	1	13.56/3.21	18.73/4.54	16.91/4.87	11.42/3.00	15.16/3.91
	BinaryViT	1	7.94/2.44	11.92/2.87	8.99/2.83	6.61/2.15	8.87/2.57
	BHViT	1	7.35/2.02	11.44/2.45	8.97/2.43	5.86/1.68	8.41/2.15
	Ours	1	5.97/2.03	10.96/2.18	8.42/2.00	4.89/1.45	**7.56/1.91**

achieves an averaged mean error of 3.14m/4.36°. Compared with the previous state-of-the-art binary neural network method, our approach reduces the error by 0.42m/0.11° while incurring no additional computational or memory cost.

Visualization. Fig. 4 and Fig. 5 show the predicted trajectories on the 8-14-14-42 subset of Oxford and the 2012-03-31 subset of NCLT, respectively. The mean position error and orientation error are also reported. Compared with other 1-bit methods, BiLoc exhibits fewer outliers and produces trajectories that are closer to its real-valued counterpart, indicating more robust localization results.

5.3 Ablation Studies

Hyper-Parameter Selection. We use the 1-bit DiffLoc framework with the BHViT backbone to determine the hyperparameters λ_1 and λ_2. Figure 6 reports the mean position error under different settings. For λ_1, which controls

Table 3: Quantitative results on the NCLT dataset. The evaluation covers the mean position(m) and orientation(°) error.

Framework	Method	Bits	2012-02-12	2012-02-19	2012-03-31	2012-05-26	Average [m/°]
PointLoc [64]	Real-valued	32	7.23/4.88	6.31/3.89	6.71/4.32	10.02/5.32	7.57/4.60
PosePN [78]	Real-valued	32	9.45/7.47	6.15/5.05	5.79/5.28	13.47/7.77	8.72/6.39
PosePN++ [78]	Real-valued	32	4.97/3.75	3.68/2.65	4.35/3.38	9.59/4.49	5.65/3.57
PoseMinkLoc [78]	Real-valued	32	6.24/5.03	4.87/3.94	4.23/4.03	10.32/6.52	6.42/4.88
PoseSOE [78]	Real-valued	32	13.09/8.05	6.16/4.51	5.24/4.56	12.60/7.67	9.27/6.20
STCLoc [77]	Real-valued	32	4.91/4.34	3.25/3.10	3.75/4.04	8.67/5.23	5.15/4.18
HypLiLoc [63]	Real-valued	32	1.71/3.56	1.68/2.69	1.52/2.90	2.90/3.47	1.95/3.16
DiffLoc [36]	Real-valued	32	0.99/2.40	0.92/2.14	0.98/2.27	1.88/2.43	1.19/2.31
	ReActNet	1	5.76/6.78	5.05/6.07	4.91/6.45	10.72/7.16	6.61/6.62
	BinaryViT	1	3.68/5.89	3.43/4.92	3.44/5.29	8.21/5.80	4.69/5.48
	BHViT	1	2.69/4.81	2.28/3.90	2.39/4.28	6.86/4.88	3.56/4.47
	Ours	1	2.15/4.71	1.95/3.73	1.94/4.15	6.52/4.87	**3.14/4.36**

the strength of feature learning from the offline real-valued counterpart, the performance improves as λ_1 increases and reaches its optimum at $\lambda_1 = 0.80$. A larger value leads to slight degradation, suggesting that overly strong supervision may hinder stable optimization. We therefore fix λ_1 to 0.80 in subsequent experiments. For λ_2, which encourages structural alignment between the 1-bit features and their real-valued counterparts, the best performance is achieved at $\lambda_2 = 0.05$. This indicates that moderate structural regularization is beneficial, while excessive alignment may restrict model flexibility. Notably, the original baseline ($\lambda_1 = 0, \lambda_2 = 0$) performs worse than all variants, confirming the effectiveness and necessity of our method.

Latency Comparison. We evaluate real-time latency on an RTX 5090 GPU for both binarized and real-valued models. Real-valued operations were executed with cuDNN [13], while binary operations are implemented using TC-BNN [34]. As shown in Table 4, the 1-bit variant achieves a 2.07× latency reduction compared with the real-valued baseline, while our method adds no inference over-

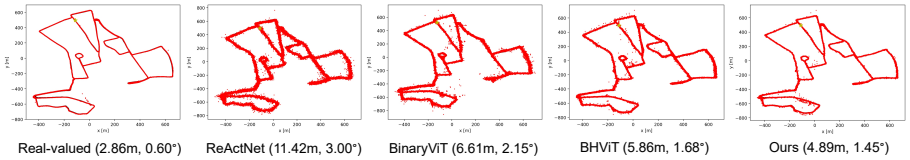

Fig. 4: LiDAR localization results on the Oxford dataset (18-14-14-42 subset). Ground truth and predictions are shown in black and red, respectively, and the star marks the first frame. Each subfigure reports the mean position and orientation errors.

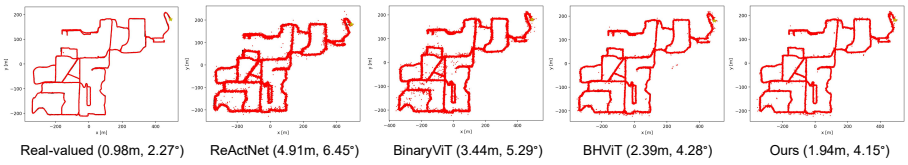

Fig. 5: LiDAR localization results on the NCLT dataset (2012-03-31 subset). Ground truth and predictions are shown in black and red, respectively, and the star marks the first frame. Each subfigure reports the mean position and orientation errors.

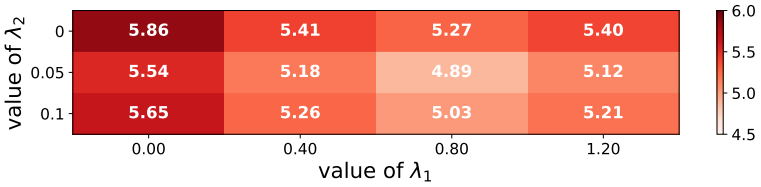

Fig. 6: The hyper-parameters λ_1 and λ_2 in our BiLoc framework are selected through empirical evaluation on the 18-14-14-42 subset of the Oxford dataset.

head. Due to the limited range of 1-bit operators supported by TC-BNN, the achieved inference speedup does not reach the theoretical upper bound. We expect that latency can be further reduced with the development of hardware accelerators specifically designed for binarized operations.

Table 4: Latency comparison in deployment (input size: 32×512).

Method	Bits	Params (M)	OPs (G)	Latency (ms)
Real-valued DiffLoc	32	39.96	77.06	30.72
1-bit DiffLoc + BHViT	1	3.01	0.17	14.83
1-bit DiffLoc + Ours	1	3.01	0.17	14.83

6 Conclusion

In this paper, we propose BiLoc, a fully binarized framework for resource-efficient outdoor LiDAR localization. By introducing an auxiliary objective that provides additional optimization guidance during training, our method enhances localization robustness and alleviates the representational limitations of BNNs. Extensive experiments on large-scale outdoor LiDAR datasets validate the effectiveness of BiLoc, establishing a new state of the art in efficient LiDAR localization and offering a practical solution for deployment on resource-constrained platforms.

7 Acknowlwdgement

This work was supported by Fundo para o Desenvolvimento das Ciências e da Tecnologia of Macau (FDCT) with Reference No. 0117/2024/RIB2.

References

1. Barnes, D., Gadd, M., Murcutt, P., Newman, P., Posner, I.: The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset. In: IEEE international conference on robotics and automation. pp. 6433–6438. IEEE (2020)
2. Belilovsky, E., Eickenberg, M., Oyallon, E.: Decoupled greedy learning of cnns. In: International Conference on Machine Learning. pp. 736–745. PMLR (2020)
3. Bengio, Y., Léonard, N., Courville, A.C.: Estimating or propagating gradients through stochastic neurons for conditional computation. ArXiv **abs/1308.3432** (2013)
4. Berger, T.: Rate Distortion Theory and Data Compression, pp. 1–39. Springer Vienna, Vienna (1975)
5. Brachmann, E., Cavallari, T., Prisacariu, V.A.: Accelerated coordinate encoding: Learning to relocalize in minutes using rgb and poses. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5044–5053 (2023)
6. Brachmann, E., Krull, A., Nowozin, S., Shotton, J., Michel, F., Gumhold, S., Rother, C.: Dsac-differentiable ransac for camera localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6684–6692 (2017)
7. Carlevaris-Bianco, N., Ushani, A.K., Eustice, R.M.: University of michigan north campus long-term vision and lidar dataset. The International Journal of Robotics Research **35**(9), 1023–1035 (2016)
8. Chen, L., Wang, D., Gan, Z., Liu, J., Henao, R., Carin, L.: Wasserstein contrastive representation distillation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16296–16305 (2021)
9. Chen, S., Li, X., Wang, Z., Prisacariu, V.: Dfnet: Enhance absolute pose regression with direct feature matching. In: Proceedings of the European Conference on Computer Vision (2022)
10. Chen, Z., Qin, H., Guo, Y., Su, X., Yuan, X., Kong, L., Zhang, Y.: Binarized diffusion model for image super-resolution. Advances in Neural Information Processing Systems **37**, 30651–30669 (2024)
11. Chen, Z., Sun, K., Yang, F., Guo, L., Tao, W.: Sc^2-pcr++: Rethinking the generation and selection for efficient and robust point cloud registration. IEEE transactions on pattern analysis and machine intelligence **45**(10), 12358–12376 (2023)
12. Cheng, X., Rao, Z., Chen, Y., Zhang, Q.: Explaining knowledge distillation by quantifying the knowledge. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12925–12935 (2020)
13. Chetlur, S., Woolley, C., Vandermersch, P., Cohen, J., Tran, J., Catanzaro, B., Shelhamer, E.: cudnn: Efficient primitives for deep learning. arXiv preprint arXiv:1410.0759 (2014)
14. Cho, J.H., Hariharan, B.: On the efficacy of knowledge distillation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4794–4802 (2019)

15. Cuturi, M.: Sinkhorn distances: Lightspeed computation of optimal transport. Advances in neural information processing systems **26** (2013)
16. Deng, S., Lin, H., Li, Y., Gu, S.: Surrogate module learning: Reduce the gradient error accumulation in training spiking neural networks. In: International Conference on Machine Learning. pp. 7645–7657. PMLR (2023)
17. Dosovitskiy, A.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021)
18. Duan, S., Principe, J.C.: Training deep architectures without end-to-end backpropagation: A survey on the provably optimal methods. IEEE Computational Intelligence Magazine **17**(4), 39–51 (2022)
19. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**, 381–395 (1981)
20. Frickenstein, A., Vemparala, M.R., Mayr, J., Nagaraja, N.S., Unger, C., Tombari, F., Stechele, W.: Binary dad-net: Binarized driveable area detection network for autonomous driving. In: IEEE International Conference on Robotics and Automation. pp. 2295–2301 (2020)
21. Gao, T., Zhang, Y., Zhang, Z., Liu, H., Yin, K., Xu, C., Kong, H.: Bhvit: Binarized hybrid vision transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3563–3572 (2025)
22. Gibson, J.D.: Rate Distortion Theory, pp. 77–92. Springer Nature Switzerland, Cham (2025)
23. Grainge, O., Milford, M., Bodala, I., Ramchurn, S.D., Ehsan, S.: Design space exploration of low-bit quantized neural networks for visual place recognition. IEEE Robotics and Automation Letters **9**(6), 5070–5077 (2024)
24. Grainge, O., Milford, M., Bodala, I., Ramchurn, S.D., Ehsan, S.: Structured pruning for efficient visual place recognition. IEEE Robotics and Automation Letters (2024)
25. He, Y., Lou, Z., Zhang, L., Liu, J., Wu, W., Zhou, H., Zhuang, B.: Bivit: Extremely compressed binary vision transformers. In: IEEE/CVF International Conference on Computer Vision. pp. 5651–5663 (2023)
26. He, Y., Zhang, X., Sun, J.: Channel pruning for accelerating very deep neural networks. In: Proceedings of the IEEE international conference on computer vision. pp. 1389–1397 (2017)
27. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
28. Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R., Bengio, Y.: Binarized neural networks. Advances in neural information processing systems **29** (2016)
29. Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., Adam, H., Kalenichenko, D.: Quantization and training of neural networks for efficient integer-arithmetic-only inference. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2704–2713 (2018)
30. Kendall, A., Cipolla, R.: Geometric loss functions for camera pose regression with deep learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5974–5983 (2017)
31. Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-dof camera relocalization. In: Proceedings of the IEEE international conference on computer vision. pp. 2938–2946 (2015)
32. Komorowski, J.: Minkloc3d: Point cloud based large-scale place recognition. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 1790–1799 (2021)

33. Le, P.H.C., Li, X.: Binaryvit: Pushing binary vision transformers towards convolutional models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 4665–4674 (June 2023)
34. Li, A., Su, S.: Accelerating binarized neural networks via bit-tensor-cores in turing gpus. IEEE Transactions on Parallel and Distributed Systems **32**(7), 1878–1891 (2020)
35. Li, W., Liu, C., Yu, S., Liu, D., Zhou, Y., Shen, S., Wen, C., Wang, C.: Lightloc: Learning outdoor lidar localization at light speed. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6680–6689 (2025)
36. Li, W., Yang, Y., Yu, S., Hu, G., Wen, C., Cheng, M., Wang, C.: Diffloc: Diffusion model for outdoor lidar localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15045–15054 (2024)
37. Li, W., Yu, S., Wang, C., Hu, G., Shen, S., Wen, C.: Sgloc: Scene geometry encoding for outdoor lidar localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9286–9295 (2023)
38. Li, Y., Xu, S., Lin, M., Cao, X., Liu, C., Sun, X., Zhang, B.: Bi-vit: Pushing the limit of vision transformer quantization. In: AAAI Conference on Artificial Intelligence. vol. 38, pp. 3243–3251 (2024)
39. Liu, S.Y., Liu, Z., Cheng, K.T.: Oscillation-free quantization for low-bit vision transformers. In: International conference on machine learning. pp. 21813–21824. PMLR (2023)
40. Liu, Z., Shen, Z., Savvides, M., Cheng, K.T.: Reactnet: Towards precise binary neural network with generalized activation functions. In: European conference on computer vision. pp. 143–159. Springer (2020)
41. Liu, Z., Wu, B., Luo, W., Yang, X., Liu, W., Cheng, K.T.: Bi-real net: Enhancing the performance of 1-bit cnns with improved representational capability and advanced training algorithm. In: Proceedings of the European conference on computer vision. pp. 722–737 (2018)
42. Luo, L., Cao, S.Y., Li, X., Xu, J., Ai, R., Yu, Z., Chen, X.: Bevplace++: Fast, robust, and lightweight lidar global localization for unmanned ground vehicles. IEEE Transactions on Robotics (2025)
43. Moreau, A., Piasco, N., Tsishkou, D., Stanciulescu, B., de La Fortelle, A.: Lens: Localization enhanced by nerf synthesis. In: Conference on Robot Learning. pp. 1347–1356. PMLR (2022)
44. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
45. Qin, H., Gong, R., Liu, X., Shen, M., Wei, Z., Yu, F., Song, J.: Forward and backward information retention for accurate binary neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2250–2259 (2020)
46. Rastegari, M., Ordonez, V., Redmon, J., Farhadi, A.: Xnor-net: Imagenet classification using binary convolutional neural networks. In: European conference on computer vision. pp. 525–542. Springer (2016)
47. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4510–4520 (2018)
48. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4938–4947 (2020)

49. Shavit, Y., Ferens, R., Keller, Y.: Coarse-to-fine multi-scene pose regression with transformers. IEEE transactions on pattern analysis and machine intelligence **45**(12), 14222–14233 (2023)
50. Shen, Z., Liu, Z., Qin, J., Huang, L., Cheng, K.T., Savvides, M.: S2-bnn: Bridging the gap between self-supervised real and 1-bit neural networks via guided distribution calibration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2165–2174 (2021)
51. Shi, C., Li, J., Gong, J., Yang, B., Zhang, G.: An improved lightweight deep neural network with knowledge distillation for local feature extraction and visual localization using images and lidar point clouds. ISPRS journal of photogrammetry and remote sensing **184**, 177–188 (2022)
52. Shwartz-Ziv, R., Tishby, N.: Opening the black box of deep neural networks via information. arXiv preprint arXiv:1703.00810 (2017)
53. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2446–2454 (2020)
54. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–9 (2015)
55. Tishby, N., Pereira, F.C., Bialek, W.: The information bottleneck method. arXiv preprint physics/0004057 (2000)
56. Tishby, N., Zaslavsky, N.: Deep learning and the information bottleneck principle. In: IEEE Information Theory Workshop. pp. 1–5 (2015)
57. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International conference on machine learning. pp. 10347–10357. PMLR (2021)
58. Uy, M.A., Lee, G.H.: Pointnetvlad: Deep point cloud based retrieval for large-scale place recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4470–4479 (2018)
59. Victor, T., Kusano, K., Gode, T., Chen, R., Schwall, M.: Safety performance of the waymo rider-only automated driving system at one million miles. Tech. Rep. (2023)
60. Villani, C., et al.: Optimal transport: old and new, vol. 338. Springer (2008)
61. Wang, B., Chen, C., Lu, C.X., Zhao, P., Trigoni, N., Markham, A.: Atloc: Attention guided camera localization. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 10393–10401 (2020)
62. Wang, J., Rupprecht, C., Novotny, D.: Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9773–9783 (2023)
63. Wang, S., Kang, Q., She, R., Wang, W., Zhao, K., Song, Y., Tay, W.P.: Hypliloc: Towards effective lidar pose regression with hyperbolic fusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5176–5185 (2023)
64. Wang, W., Wang, B., Zhao, P., Chen, C., Clark, R., Yang, B., Markham, A., Trigoni, N.: Pointloc: Deep pose regressor for lidar point cloud localization. IEEE Sensors Journal **22**(1), 959–968 (2021)
65. Wang, Y., Solomon, J.M.: Deep closest point: Learning representations for point cloud registration. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3523–3532 (2019)

66. Wang, Y., Ni, Z., Pu, Y., Zhou, C., Ying, J., Song, S., Huang, G.: Infopro: Locally supervised deep learning by maximizing information propagation. International Journal of Computer Vision **133**(5), 2752–2782 (2025)
67. Wang, Z., Wu, Z., Lu, J., Zhou, J.: Bidet: An efficient binarized object detector. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition pp. 2046–2055 (2020)
68. Wolcott, R.W., Eustice, R.M.: Fast lidar localization using multiresolution gaussian mixture maps. In: 2015 IEEE international conference on robotics and automation (ICRA). pp. 2814–2821. IEEE (2015)
69. Xu, S., Li, Y., Zeng, B., Ma, T., Zhang, B., Cao, X., Gao, P., Lü, J.: Ida-det: An information discrepancy-aware distillation for 1-bit detectors. In: European Conference on Computer Vision. pp. 346–361. Springer (2022)
70. Xu, S., Wang, M., Li, Y., Lin, M., Zhang, B., Doermann, D., Sun, X.: Learning 1-bit tiny object detector with discriminative feature refinement. In: Forty-first International Conference on Machine Learning (2024)
71. Yang, B., Li, Z., Li, W., Cai, Z., Wen, C., Zang, Y., Muller, M., Wang, C.: Lisa: Lidar localization with semantic awareness. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15271–15280 (2024)
72. Yang, Y., Li, W., Ao, S., Xu, Q., Yu, S., Guo, Y., Zhou, Y., Shen, S., Wang, C.: Raloc: Enhancing outdoor lidar localization via rotation awareness. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3304–3313 (2025)
73. Yin, H., Xu, X., Lu, S., Chen, X., Xiong, R., Shen, S., Stachniss, C., Wang, Y.: A survey on global lidar localization: Challenges, advances and open problems. International Journal of Computer Vision **132**, 3139 – 3171 (2024)
74. Yin, K., Gao, T., Kong, H.: Pathfinder for low-altitude aircraft with binary neural network. In: IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 20692–20699 (2025)
75. Yin, K., Zhang, Z., Kong, S., Gao, T., Xu, C.Z., Kong, H.: Information-bottleneck driven binary neural network for change detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7176–7186 (2025)
76. Yin, P., Zhu, X., Song, J., Gao, L., Shen, H.T.: Si-bivit: Binarizing vision transformers with spatial interaction. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 8169–8178 (2024)
77. Yu, S., Wang, C., Lin, Y., Wen, C., Cheng, M., Hu, G.: Stcloc: Deep lidar localization with spatio-temporal constraints. IEEE Transactions on Intelligent Transportation Systems **24**(1), 489–500 (2022)
78. Yu, S., Wang, C., Wen, C., Cheng, M., Liu, M., Zhang, Z., Li, X.: Lidar-based localization using universal encoding and memory-aware regression. Pattern Recognition **128**, 108685 (2022)
79. Zhang, L., Song, J., Gao, A., Chen, J., Bao, C., Ma, K.: Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3713–3722 (2019)
80. Zhang, Q., Cheng, X., Chen, Y., Rao, Z.: Quantifying the knowledge in a dnn to explain knowledge distillation for classification. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(4), 5099–5113 (2022)