

FeDepth: Federated Learning for Depth Estimation under Robot Heterogeneity

Ganghyeon Lee^{*}, Inha Lee^{*}, Junhee Lee, Jeongeon Lee,
Sung Whan Yoon, and Kyungdon Joo[†]

Ulsan National Institute of Science and Technology (UNIST), Republic of Korea
`{create0327,epsilon8854,junhee98,aeonian,shyoon8,kyungdon}@unist.ac.kr`

Abstract. Although recent robot perception research emphasizes training on data from diverse environments to improve generalization, most existing methods still rely on centralized learning, which is inefficient and difficult to scale across heterogeneous robot platforms. Federated learning (FL) offers an alternative by enabling distributed training without raw data transfer, but it suffers from severe performance degradation under domain shifts caused by heterogeneity across clients. In real robotic deployments, data distributions often overlap across platforms, environments, and sensing conditions, making it difficult to partition clients into clearly separated domains. However, this characteristic breaks the assumption of clearly separable client domains commonly used in clustered FL. To address this gap in robot perception, particularly in depth estimation, we introduce two realistic and unexplored non-IID scenarios that reflect heterogeneity in terms of platform, environment, and depth distribution. We then propose FEDEPTH, a descriptor-based clustered FL framework that models client relationships through soft clustering. Unlike hard clustering methods that assume clearly separated clusters, FEDEPTH allows clients to participate in multiple clusters, capturing continuous and ambiguous domain transitions commonly observed in robotic environments. Extensive experiments demonstrate that FEDEPTH consistently improves robustness over standard FL and clustered FL baselines across multiple depth estimation architectures, providing a practical and effective solution for federated robot perception. Our project page is available at <https://vision3d-lab.github.io/fedepth/>.

Keywords: Federated Learning · Robot Perception · Depth Estimation

1 Introduction

Robot perception, including tasks such as depth estimation [3], object recognition [36], and semantic understanding [8], is fundamental to enabling autonomous systems to interact safely and intelligently with their surroundings. Recent advances in deep learning, particularly convolutional neural networks (CNNs) and

^{*} Equal contribution. [†] Corresponding author.

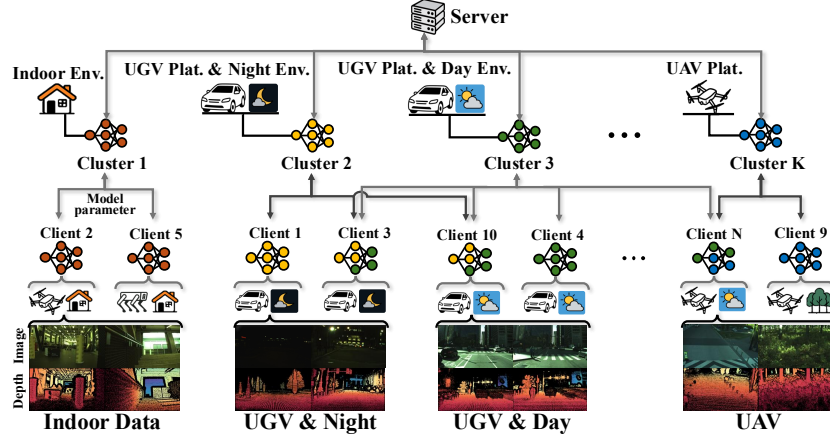


Fig. 1: Illustration of FEDEPTH. The proposed FEDEPTH, a soft clustering-based FL framework that models complex inter-client relationships for robust training and effective cluster model updates. For example, Clients 3 and 10, which share the same platform (*e.g.*, UGV), are assigned to two overlapping clusters.

Transformer-based architectures, have notably improved perception performance across various modalities [13, 32, 45]. In addition, training on datasets collected from diverse environments [22, 42] has been shown to enhance generalization beyond controlled laboratory conditions to the complex and variable settings encountered in real-world robotic applications.

Despite these advances, most perception pipelines still rely on a centralized learning (CL) paradigm: (*i*) collecting raw sensory data in the field, (*ii*) transmitting it to a central server, and (*iii*) training a single model on the aggregated dataset. Although this approach has proven effective in controlled benchmarks with curated data, it becomes impractical when training on data acquired from real-world robots. Concretely, CL suffers from several limitations, including high communication overhead incurred by transmitting high-bandwidth sensory data (*e.g.*, high-resolution images or LiDAR streams) to a central server. Furthermore, CL faces severe scalability bottlenecks, as the computational requirements of servers scale proportionally with the volume of the aggregated dataset. Moreover, CL is highly vulnerable to increasing privacy concerns when collecting visual perception data from personal robotic agents.

Federated learning (FL) [31] provides a promising alternative: each robot trains its model locally and shares only weights rather than raw images. Although FL is widely known to perform well for canonical image classification benchmarks (*e.g.*, CIFAR-100 [24]), our preliminary experiments reveal that FL in robot perception scenarios inherently suffers from severe heterogeneity, which is driven by the diversity of robot platforms (*e.g.*, UGV, UAV, legged robots), various observation environments (*e.g.*, urban, forest, indoor), and unbalanced data quantities across clients, thereby raising a significant challenge to deploying

FL in real-world robots. It should be noted that only a few recent works have pushed FL to robot perception applications [23, 29, 33]. However, these studies are confined to a single platform (*e.g.*, manipulator) or a limited environment (*e.g.*, on-road driving), thus failing to address the substantial domain gaps induced by heterogeneous real-world robots. Moreover, data distributions across robots are often partially overlapping rather than clearly separable, as robots may share similar environments, sensing conditions, or platform characteristics, suggesting the need for soft clustering that allows overlapping client assignments.

In this work, we focus on two real-world scenarios that encompass the domain shifts frequently encountered in heterogeneous robot data. First, a heterogeneous platform-environment (HPE) scenario incorporates the heterogeneity originating from a unique combination of platform and environment for each client’s trajectory. The combinatorial configurations induce variations in viewpoint, motion dynamics, and scene layout, thus leading to significant domain gaps that hinder federated optimization. Second, a bi-modal range (BMR) scenario contains a bi-modal distribution over depth ranges, distinctly separated into indoor and outdoor domains. Indoor clients observe close-range geometry (under 10 meters), while outdoor clients capture long-range scenes (up to 80 meters), creating a fundamental mismatch in depth distribution. This divergence, coupled with differing camera intrinsics, severely limits the performance of unified models even under centralized training.

To address these heterogeneities, we propose FEDEPTH, a clustered-FL (CFL) framework for monocular depth estimation on heterogeneous multi-robot systems (see Fig. 1). In FEDEPTH, we effectively tackle heterogeneous data by grouping clients based on their data distribution and performing intra-group aggregation. This approach ensures that each cluster consists of clients with relatively homogeneous data, which facilitates robust federated optimization even under heterogeneity. This principle is rooted in CFL [38], and we observe that a CFL paradigm can mitigate the challenges of robot-driven heterogeneity. However, these conventional CFL approaches do not yield stable performance gains across all baselines. Furthermore, clients in real-world scenarios often share multiple attributes, making it suboptimal or even infeasible to assign a client to a single, unique group. In contrast, FEDEPTH permits a client to be associated with multiple groups, enabling a single client to contribute to multiple relevant clusters. To this end, we also propose a client clustering algorithm capable of performing this multi-cluster assignment. We apply FEDEPTH to depth estimation, which is a fundamental robot perception task that spans diverse architectures and representations (*e.g.*, bins [3], frequency [41]). We evaluate FEDEPTH on the HPE and BMR scenarios, where FEDEPTH generalizes across datasets as well as depth estimation models. Our main contributions are:

- We systematically investigate the impact of robot-driven heterogeneity on FL, a critical but under-explored challenge in robot perception.
- We design two novel non-IID scenarios (HPE and BMR) derived from robot perception datasets. These scenarios establish a new realistic testbed for benchmarking federated depth estimation under robot-driven heterogeneity.

- We propose FEDEPTH, a CFL framework that resolves robot-driven heterogeneity using soft cluster assignments. We can handle overlapping client distributions that conventional CFL methods cannot effectively manage.
- We validate that FEDEPTH is model-agnostic, demonstrating generalization by integrating it with diverse depth estimation networks and representations.

2 Related Work

2.1 Federated Learning

Federated learning (FL) enables multiple clients to collaboratively train a model in a distributed environment without sharing raw data. A representative for FL is FedAvg [31]. Unlike CL, FL typically operates under non-IID data, where client data follow different distributions, resulting in heterogeneity across clients. Such heterogeneity often leads to degraded convergence, reduced global model accuracy, and performance imbalance across clients [28, 47]. To address these challenges, several methods [1, 21, 27] mitigate client drift through proximity constraints, control variates, or regularized local objectives. However, these approaches still assume that a single global model can effectively serve all clients despite their heterogeneous data distributions [38]. Personalized federated learning (pFL) methods [10, 11, 48] alleviate this limitation by learning client-specific models through local personalization. However, they typically assume a closed-client setting, where inference data originates from training clients, limiting applicability to unseen clients or sequences.

In contrast, clustered federated learning (CFL) methods address data heterogeneity by grouping clients with similar data distributions and training specialized models for each cluster. Early studies include CFL [38] and IFCA [16], which partition clients into clusters and jointly optimize cluster-specific models to handle heterogeneous data distributions. Building on this idea, several one-shot clustering approaches have been proposed, including PACFL [40], which leverages principal angles for clustering, and FedClust [19], which constructs clusters based on model parameters. Beyond clustering-based approaches, HCFL [18] introduces an integrated framework that combines clustering with hierarchical model aggregation, while LCFed [46] further extends this direction by jointly optimizing model partitioning and global-local aggregation.

2.2 Applications in Robot Perception

Recently, FL has been adopted in robot perception [23, 29, 33]. For instance, using an exponential moving average (EMA) of global models, FedEMA [23] addresses temporal catastrophic forgetting in street scene semantic understanding. In object detection, FedVision [29] enables collaborative learning by allowing multiple institutions to perform data annotation and model training locally on their images without sharing the data. On the other hand, stereo matching leverages FL to enable models to collaboratively adapt to diverse and challenging environments in real-time, improving accuracy on resource-constrained devices without

sharing raw data [33]. These approaches still rely on a single shared model, which makes it challenging to effectively handle the diverse domains encountered in robotic settings.

2.3 Monocular Depth Estimation

Monocular depth estimation research has continued to develop since the introduction of CNN-based deep learning approaches. Following the seminal work [12], which first applies CNNs to monocular depth estimation, early studies primarily focus on encoder-decoder architectures [25], formulating depth estimation as a regression problem. Subsequently, CNN-based classification approaches that discretize depth values into bins [6] and bin-based methods [3, 4, 26] leveraging CNNs emerge, improving performance and robustness. Additionally, Transformer-based approaches [35] achieve notable improvements over CNNs by leveraging the global attention mechanism of Vision Transformers. Recently, foundation model-based approaches [42, 43] demonstrate strong zero-shot generalization capabilities across diverse environments.

FedSCDepth [37] represents one of the pioneering efforts to integrate FL with self-supervised depth estimation for autonomous vehicles. However, evaluations primarily focus on constrained datasets (*e.g.*, KITTI [14], DDAD [17]), which provide only limited coverage of diversity encountered in real-world scenarios. In contrast, our work systematically accounts for the non-IID conditions, including multi-platform and environmental heterogeneity, as well as bi-modal depth range spanning indoor and outdoor scenes.

3 Method

In this section, we present the design of FEDEPTH. Sec. 3.1 formalizes the federated depth estimation task and outlines the optimization issues under robot data heterogeneity. Sec. 3.2 introduces two real-world robot heterogeneity scenarios that encompass depth range, platform, and environment shifts. To address these challenges, Sec. 3.3 presents FEDEPTH, a new soft clustering-based FL framework that carefully models the complex relationships between clients, thereby ensuring robust training and effective cluster model updates.

3.1 Problem Statement

Problem. We address a core challenge in robot perception: training a monocular depth estimation model across distributed, heterogeneous robots in an FL setup. In this setting, each client collects and processes data independently under varying conditions. The goal is to collaboratively learn global depth models that generalize across diverse robot platforms without centralized data collection.

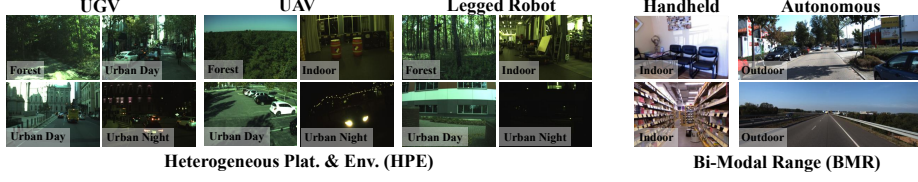


Fig. 2: Overview of the two scenarios for robot heterogeneity. (Left) HPE introduces shifts across platforms and environments (M3ED [7]). (Right) BMR reflects bi-modal depth distributions between indoor (NYUv2 [39]) and outdoor (KITTI [15]).

Formal Definition. Let $\mathcal{X} = \mathbb{R}^{H \times W \times 3}$ denote the space of RGB images and $\mathcal{Y} = \mathbb{R}^{H \times W}$ the space of depth maps. Given an input image $\mathbf{x} \in \mathcal{X}$, FL aims to learn a model $f_w : \mathcal{X} \rightarrow \mathcal{Y}$ with learnable parameter w . Training is supervised using a loss function $\mathcal{L}(\mathbf{y}, \mathbf{y}^*)$, such as pixel-wise ℓ_1 , ℓ_2 , or scale-invariant loss [12], where $\mathbf{y} = f_w(\mathbf{x})$ is the predicted depth map and $\mathbf{y}^*$ is the corresponding ground truth. In this setting, data is distributed across a set of clients $\mathcal{C}$, where each client $c \in \mathcal{C}$ holds its own local dataset $\mathcal{X}_c$. The global training objective is to minimize the population loss:

$$\begin{aligned} \min_w F(w) &\triangleq \sum_{c \in \mathcal{C}} p_c F_c(w), \\ F_c(w) &\triangleq \mathbb{E}_{(\mathbf{x}, \mathbf{y}^*) \sim \mathcal{X}_c} [\mathcal{L}(\mathbf{y}, \mathbf{y}^*)]. \end{aligned} \quad (1)$$

Here, $F_c(w)$ is the local objective function for client c , and $F(w)$ is the weighted average over these with $p_c = \frac{|\mathcal{X}_c|}{\sum_j |\mathcal{X}_j|}$, as formulated in FedAvg [31] algorithm.

Challenges in Federated Depth Estimation. Monocular depth estimation learns the mapping from RGB images to depth maps by modeling the underlying depth distribution. Yet its depth distribution depends on the acquisition setup, including the environment, robot platform, and sensor configuration. Such heterogeneity across these setups makes client data increasingly non-IID, leading to domain shifts. To address these challenges, we pursue an FL strategy that explicitly models the domain heterogeneity across clients. Rather than assuming that a single global model can effectively capture all data distributions, we aim to uncover structure within the client population and group clients with similar characteristics. This enables the training of specialized models that are better aligned with each domain.

3.2 Robot Heterogeneity Scenarios

To examine the effects of heterogeneity in federated depth estimation, we introduce two real-world scenarios called HPE and BMR that reflect distinct and complementary sources of domain shift (see Fig. 2). These settings are designed

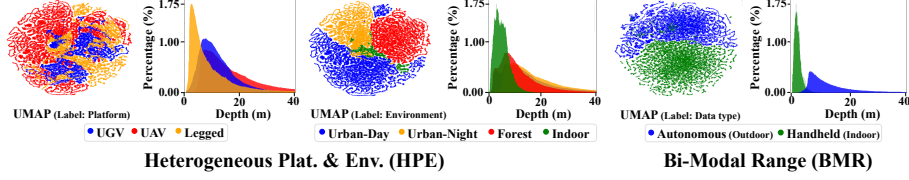


Fig. 3: Data characteristics of two scenarios. Depth distribution and UMAP visualization [30].

to isolate the impact of (i) differences in platform and environment combinations, and (ii) differences in depth distribution across indoor and outdoor scenes. Moreover, our scenarios are designed to model a spectrum of inter-client relationships, ranging from cases where clients are clearly separated into distinct distributions to cases where clients are difficult to distinguish due to overlapping distributions and multiple attributes. By modeling such realistic variations, we establish a controlled yet challenging testbed for evaluating the robustness and adaptability of FL strategies under non-IID conditions. We construct the proposed two scenarios by curating existing datasets: M3ED [7] for the HPE scenario, NYUv2 [39] and KITTI [15] for the BMR scenario.

Heterogeneous Platform-Environment (HPE) Scenario. HPE Scenario contains various platforms (*e.g.*, UGV, UAV, and legged robot) and environments (*e.g.*, urban day/night, forest, and indoor). Each client is assigned a single continuous trajectory to reflect realistic robot deployments. Environmental variations influence both scene geometry (*e.g.*, planar indoor structures *vs.* irregular outdoor foliage) and appearance characteristics (*e.g.*, shape and texture distributions). Furthermore, these platforms and environment attributes are often combinatorial and overlapping across clients (see the left of Fig. 3). For instance, Client A (Legged robot) may capture data in a ‘Forest’ environment, while Client B (UAV) operates in the same ‘Forest’. While both clients share the ‘Forest’ attribute, their data distributions diverge due to platform-specific characteristics like viewpoint and motion dynamics. This combinatorial nature and the presence of multiple, overlapping attributes make it challenging to assign a client to a single, unique group. This ambiguity undermines the clear cluster structures assumed by conventional CFL methods and leads to performance degradation under severe data heterogeneity.

Bi-Modal Range (BMR) Scenario. The BMR scenario models a severe form of heterogeneity where clients are divided into two fundamentally distinct groups: indoor and outdoor. Clients consist of indoor images (*e.g.*, residence interiors and market aisles) and outdoor images (*e.g.*, roads and urban environments). Indoor clients observe enclosed, close-range scenes (under 10 meters), while outdoor clients capture open, far-range environments (up to 80 meters). This difference leads to different depth distributions, which are further amplified

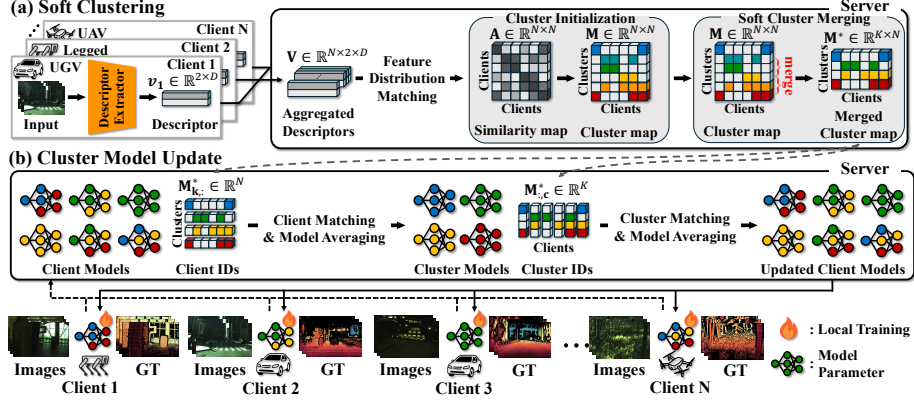


Fig. 4: Overall framework of FEDEPTH. (a) The Soft Clustering stage is first performed before the initial round. In this stage, each client utilizes a Descriptor Extractor to generate a client descriptor summarizing its local data distribution. Then, the server aggregates these descriptors and performs clustering to determine a cluster map. Second, (b) in the Cluster Model Update stage, clients conduct iterative local training in each round. At the end of each round, the server aggregates the client models, updates the cluster models, and propagates them according to the cluster map.

by discrepancies in camera parameters such as resolution, focal length, and aspect ratio. These combined factors result in fundamentally different supervision signals across clients (see the right of Fig. 3), allowing for the evaluation of how range heterogeneity impacts FL.

3.3 Federated Learning for Depth Estimation

Based on the two scenarios, FEDEPTH presents a novel CFL approach to address the severe robotic heterogeneity (see Fig. 4). FEDEPTH first performs the *Soft Clustering* stage, which aims to allocate each client into appropriate clusters by considering its robotic settings. Specifically, we utilize visual descriptors, which are extracted by a pre-trained encoder to model the data distribution of each client. Based on the pairwise similarity between these descriptors, FEDEPTH constructs a soft cluster map that captures complex inter-client relationships. This design allows each client to participate in multiple clusters, thereby modeling partially overlapping data distributions that frequently arise in heterogeneous robotic deployments. In the *Cluster Model Update* stage, clients upload their locally-trained models, and the server performs cluster-wise model aggregation for each cluster. Finally, the server sends each client an updated model obtained by averaging the models of the clusters to which the client belongs. In the *Inference* stage, each unseen sequence is assigned to the relevant cluster models based on similarity and the clustering threshold.

Soft Clustering. In robotic settings, clustering based on image distributions that capture variations in motion or environment is closely related to visual place recognition [2, 34]. Inspired by this, we propose a descriptor-based soft clustering scheme that leverages the characteristics of feature distribution (see Fig. 4a). To represent the feature distribution of each client, we employ a descriptor extractor to obtain a client-level descriptor. For each client c , we randomly sample a subset $\mathcal{X}'_c \subseteq \mathcal{X}_c$ of size $n_c = \min(n, |\mathcal{X}_c|)$ and extract the feature map for each image $\mathbf{x}_c^i \in \mathcal{X}'_c$:

$$z_c^i = f_{\text{enc}}(\mathbf{x}_c^i) \in \mathbb{R}^{H' \times W' \times D},$$

where $f_{\text{enc}}(\cdot)$ denotes the pre-trained encoder of the respective baseline models (*e.g.*, AdaBins [3], NeWCRFs [44], and DCDepth [41]). Then, to effectively compress this feature, we utilize generalized mean (GeM) pooling [34]:

$$g_c^i = \left(\frac{1}{H'W'} \sum_h \sum_w (z_c^i(h, w))^p \right)^{\frac{1}{p}}, \quad p > 0. \quad (2)$$

We aggregate g_c^i into an aggregated descriptor $\mathbf{G}_c$, and compute its mean μ_c and variance σ_c^2 to represent the feature distribution of each client. The pair (μ_c, σ_c^2) defines the client descriptor v_c , capturing both the central tendency and variability of client features, enabling robust comparison between heterogeneous clients. The resulting descriptor is then transmitted to the server. On the server side, all aggregated client descriptors $\mathbf{V} = \{v_c\}_{c=1}^N$ are utilized to perform clustering suitable for the robotic setting. To assess the similarity between client domains, we approximate each descriptor with a Gaussian distribution with diagonal covariance for computational efficiency and perform feature distribution matching. Although the discrepancy between two Gaussian distributions can be measured using the KL divergence, it is directional and depends on the order of the two distributions. To obtain an order-invariant measure, we employ the Jeffreys divergence [20], a symmetric bidirectional form of the KL divergence, which is defined as follows:

$$\mathbf{A}_{i,j} = D_{\text{KL}}(\mathcal{N}(\mu_i, \sigma_i^2) \parallel \mathcal{N}(\mu_j, \sigma_j^2)) + D_{\text{KL}}(\mathcal{N}(\mu_j, \sigma_j^2) \parallel \mathcal{N}(\mu_i, \sigma_i^2)), \quad (3)$$

where $\mathcal{N}(\cdot, \cdot)$ denotes the Gaussian distribution of each client, and $D_{\text{KL}}(\cdot \parallel \cdot)$ is the KL divergence. Using Eq. (3), we construct the similarity map $\mathbf{A} \in \mathbb{R}^{N \times N}$, where N denotes the number of clients, and each entry $\mathbf{A}_{i,j}$ represents the pairwise similarity between clients. However, due to robot heterogeneity, the domains of clients are intricately mixed, making it difficult to clearly separate clusters using this naive similarity map with simple criteria.

To handle this complexity, we employ a soft clustering strategy, which allows a client to belong to multiple clusters. Concretely, we first generate a binary initial cluster map $\mathbf{M} \in \{0, 1\}^{N \times N}$ to group clients with high similarity based on a clustering threshold τ . This thresholding step effectively identifies strongly related clients while still allowing overlapping memberships across clusters. Here, each row $\mathbf{M}_{k,:}$ corresponds to the set of clients assigned to cluster k . By adjusting

the threshold τ , the clustering behavior can be dynamically controlled, ranging from highly overlapping soft clusters to strictly disjoint hard clusters.

Given the initial cluster map $\mathbf{M}$, we perform a soft cluster merging step to resolve redundant partitions, as several preliminary clusters exhibit highly correlated feature distributions. For any pair of clusters $\mathbf{M}_{i,:}$ and $\mathbf{M}_{j,:}$, if one cluster is a subset of the other, the subset cluster is merged into the dominant cluster. The merging step iterates until convergence, resulting in the final merged cluster map $\mathbf{M}^* \in \{0, 1\}^{K \times N}$, where K denotes the number of clusters, each row $\mathbf{M}_{k,:}^*$ encodes the client IDs belonging to cluster k , while each column $\mathbf{M}_{:,c}^*$ encodes the cluster IDs associated with client c (see Fig. 6).

Cluster Model Update. As shown in Fig. 4b, built upon the FL formulation in Eq. (1), FEDEPTH runs repetitive rounds of local training and aggregation. For the cluster-wise aggregation, each cluster k aggregates the locally trained models from its component clients by client IDs $\mathbf{M}_{k,:}^*$ to compute the model for cluster k at round t , *i.e.*, w_k^t . On the client side, each client c downloads the associated cluster models by cluster IDs $\mathbf{M}_{:,c}^*$ and takes the average of them to initialize the local training of the next round, *i.e.*, w_c^{t+1} :

$$w_k^t \leftarrow \sum_{c \in \mathbf{M}_{k,:}^*} p_c^k w_c^t, \quad w_c^{t+1} \leftarrow \frac{1}{|\mathbf{M}_{:,c}^*|} \sum_{k \in \mathbf{M}_{:,c}^*} w_k^t, \quad (4)$$

where $p_c^k = \frac{|\mathcal{X}_c|}{\sum_{j \in \mathbf{M}_{k,:}^*} |\mathcal{X}_j|}$ is the data-size weight in cluster k , as in FedAvg [31].

Inference. At inference time, we extract the descriptor v_{test} of a test sequence using the same frozen encoder as in the clustering stage. The similarity between the test descriptor and cluster k is computed as

$$S_{\text{test}}^k = \frac{1}{|\mathbf{M}_{k,:}^*|} \sum_{c \in \mathbf{M}_{k,:}^*} \mathbf{A}_{\text{test},c}, \quad (5)$$

where $\mathbf{A}_{i,j}$ denotes the Jeffreys divergence defined in Eq. (3). Following the soft clustering step, we apply the same threshold τ and define the candidate cluster set as $\mathcal{K}_{\text{test}} = \{k \mid S_{\text{test}}^k \leq \tau\}$. The prediction model is obtained by averaging the selected cluster models, consistent with Eq. (4). If no cluster satisfies the threshold, the test sequence is assigned to $\arg \min_k S_{\text{test}}^k$. Cluster selection is performed once per sequence, after which the selected cluster model is used to perform depth estimation for the remaining frames of the sequence. The full FEDEPTH procedure is summarized in the supplementary material.

4 Experiments

4.1 Experimental Setup

To validate the proposed method, we apply CL, FL, CFL, and FEDEPTH to various monocular depth estimation baselines under the HPE and BMR scenarios. Due to the high computational cost of federated depth estimation (2–4

days per experiment using four RTX 4090 GPUs), we follow standard practice in monocular depth estimation [3, 41, 44]. Results are reported as the mean over the last five rounds for stability, with standard deviations provided in the supplementary material. In HPE, the clustering threshold τ is selected to yield an overlap ratio of 20%. For the BMR scenario, we set τ such that the resulting cluster assignments closely approximate those of the hard clustering method to explicitly verify the effectiveness of the clustering.

Datasets. In the HPE scenario, we employ M3ED [7] and construct a train/test split that maximizes coverage of platform-environment combinations. Specifically, we split 41 sequences for training and include 11 unseen sequences in the test set to ensure evaluation under diverse conditions. Each sequence is treated as a separate client (41 clients total) and is annotated by environment type: Indoor, Urban-Day, Urban-Night, or Forest. Full details of the split and annotation are provided in the supplementary material.

In the BMR scenario, we use NYUv2 [39] and KITTI [15] as two distinct data distributions. We construct 32 clients, including 15 clients from NYUv2 and 17 clients from KITTI. Each client performs training on a single dataset. We follow the official NYUv2 train/test split and adopt the KITTI split proposed by [12].

Baselines. The choice of model architecture can influence the performance of FL. Thus, we provide a comprehensive evaluation by benchmarking combinations of various depth estimation models and representative FL algorithms. Specifically, we cover diverse depth estimation baseline architectures, such as AdaBins [3], NeWCRFs [44], DCDepth [41], and follow the standard robotics protocol of testing generalization on unseen clients (*i.e.*, data from clients not encountered during training). For FL algorithms, we benchmark FedAvg [31], as well as methods designed to mitigate heterogeneity, including FedProx [27], FedDyn [1], SCAFFOLD [21], in addition to representative CFLs [19, 40]. Given the focus on generalization to unseen clients in robotic settings, we include representative CFL methods designed for such scenarios. In particular, FedClust [19] performs hard clustering based on model parameters, and PACFL [40] applies hard clustering using image representations. To accommodate high-resolution robotic data, we modify PACFL by replacing raw images with our descriptor extractor, and use this adapted version as PACFL[†]. As pFL methods [10, 11, 48] typically involve client-side adaptation at inference time, which differs from our evaluation protocol without test-time updates, we report additional comparisons in the supplementary material.

4.2 Evaluations

HPE Scenario. Due to the robot-driven heterogeneity in the HPE scenario, FL algorithms exhibit performance degradation compared to CL, even those explicitly designed to handle non-IID data (*e.g.*, FedProx, FedDyn, and SCAFFOLD), as shown in Tab. 1. While feature-based CFL methods like PACFL[†] form reasonable clusters and thereby show improvements over standard FL,

Table 1: Quantitative evaluation on the HPE scenario. Methods are categorized into CL, FL methods (FedAvg [31], FedProx [27], FedDyn [1], SCAFFOLD [21]), CFL approaches (FedClust [19], PACFL [40]), and the proposed framework (Ours). PACFL[†] denotes the modified version of PACFL that uses our descriptor extractor. The best result per metric for each baseline is **bold**, and the second best is underlined.

Baseline	Method	$Abs\ Rel \downarrow$	$Sq\ Rel \downarrow$	$RMSE \downarrow$	$RMSE_{log} \downarrow$	$ \delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$
AdaBins [3]	CL	0.222	1.477	3.552	0.254	0.733	0.887	0.941
	FedAvg	<u>0.434</u>	<u>2.990</u>	6.959	0.542	0.369	0.580	0.729
	FedProx	0.439	3.029	<u>6.933</u>	0.533	0.369	0.585	0.737
	FedDyn	0.440	3.063	7.027	0.551	0.367	0.573	0.721
	SCAFFOLD	0.631	6.009	9.664	0.749	0.225	0.432	0.603
	FedClust	0.444	3.263	7.521	0.566	0.277	0.550	0.733
	PACFL [†]	0.602	6.447	7.348	<u>0.509</u>	<u>0.372</u>	<u>0.605</u>	<u>0.768</u>
	Ours	0.318	2.127	5.354	0.364	0.515	0.770	0.897
NeWCRFs [44]	CL	0.166	0.994	2.987	0.190	0.827	0.935	0.968
	FedAvg	0.366	1.994	4.475	0.406	0.504	0.717	0.834
	FedProx	0.364	1.983	4.479	0.405	0.504	0.718	0.834
	FedDyn	0.363	1.957	<u>4.473</u>	0.404	0.505	0.719	0.834
	SCAFFOLD	0.356	2.060	4.569	0.376	0.498	0.726	0.864
	FedClust	0.487	4.325	7.760	0.615	0.241	0.507	0.701
	PACFL [†]	<u>0.318</u>	<u>1.934</u>	4.265	<u>0.327</u>	<u>0.540</u>	<u>0.791</u>	<u>0.928</u>
	Ours	0.249	1.750	3.838	0.261	0.703	0.892	0.954
DCDepth [41]	CL	0.159	0.962	2.916	0.183	0.835	0.939	0.971
	FedAvg	0.351	2.077	4.640	0.348	0.493	0.766	0.902
	FedProx	0.354	2.152	4.641	0.348	0.492	0.769	0.902
	FedDyn	0.352	2.109	4.699	0.349	0.491	0.767	0.903
	SCAFFOLD	0.329	2.050	4.880	0.350	0.472	0.759	0.914
	FedClust	0.349	3.132	6.423	0.434	0.498	0.712	0.815
	PACFL [†]	<u>0.321</u>	<u>1.976</u>	4.338	<u>0.325</u>	<u>0.537</u>	<u>0.796</u>	<u>0.924</u>
	Ours	0.293	1.872	<u>4.469</u>	0.310	0.547	0.840	0.941

their performance is inherently limited by the hard clustering representation when dealing with complex, overlapping distributions. Moreover, this limitation becomes more pronounced under inaccurate cluster assignments, as observed with AdaBins trained via FedClust. In contrast, FEDEPTH effectively represents the complex and overlapping data distributions inherent in robotic perception by utilizing soft cluster assignments. Consequently, as shown in Tab. 1, and Fig. 5, FEDEPTH consistently outperforms all other FL baselines across all evaluated depth estimation architectures. The convergence curves in Fig. 6 show faster and more stable convergence. This demonstrates that FEDEPTH provides an effective and model-agnostic framework for FL under complex robot-driven heterogeneity.

BMR Scenario. The BMR scenario consists of two clearly separated and non-overlapping domains (indoor/outdoor), providing an ideal setting for conventional hard clustering methods such as PACFL[†]. As shown in Tab. 2, FedAvg still suffers severe performance degradation, confirming that even simple domain heterogeneity is a challenge. In particular, as reported in prior work [5], bin-based methods such as AdaBins may struggle even under CL. Training a depth distribution (bins) with a single model on a mixed dataset with clearly distinct

Table 2: Quantitative evaluation on the BMR scenario. Additional quantitative results are available in the supplementary material.

Baseline	Method	$Abs\ Rel \downarrow$	$Sq\ Rel \downarrow$	$RMSE \downarrow$	$RMSE_{log} \downarrow$	$ \delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$
AdaBins [3]	CL	0.166	0.369	2.073	0.191	0.768	0.945	0.984
	FedAvg	0.357	1.153	3.329	0.409	0.211	0.673	0.913
	PACFL [†]	0.124	0.263	2.022	0.164	0.831	0.970	0.994
	Ours	<u>0.126</u>	<u>0.283</u>	<u>2.150</u>	<u>0.167</u>	<u>0.827</u>	<u>0.968</u>	0.994
NeWCRFs [44]	CL	0.077	0.100	1.209	0.100	0.950	0.994	0.998
	FedAvg	0.323	1.048	3.094	0.344	0.191	0.833	0.983
	PACFL [†]	0.087	0.145	<u>1.501</u>	0.117	0.921	0.991	0.999
	Ours	<u>0.089</u>	0.145	1.499	<u>0.118</u>	0.921	0.991	0.999
DCDepth [41]	CL	0.074	0.101	1.246	0.099	0.952	0.994	0.998
	FedAvg	0.158	0.265	1.789	0.183	0.801	0.969	0.996
	PACFL [†]	0.082	<u>0.130</u>	1.428	<u>0.109</u>	0.933	0.993	0.999
	Ours	0.082	0.127	<u>1.430</u>	0.108	0.933	0.993	0.999

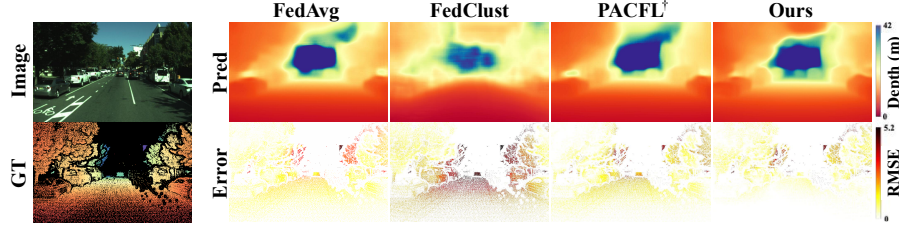


Fig. 5: Qualitative results for DCDepth on the HPE scenario. (Left) Input image and Ground Truth (GT). (Right) Prediction maps (1st row) and error (RMSE) maps (2nd row) for our method (Ours) and the baselines (FedAvg, FedClust, PACFL[†]).

depth ranges (*e.g.*, 10m vs 80m) disrupts the learned bin distribution. In contrast, CFL mitigates this issue by grouping similar clients. Our results show that FEDEPTH is not limited to overlapping domains and can naturally form well-defined clusters when data distributions are clearly separated. For NeWCRFs and DCDepth, near-binary cluster maps are formed (see Fig. 6), achieving performance comparable to PACFL[†]. For AdaBins, feature representations are less sharply separable, preventing a perfectly binary split. Consequently, although FeDepth still outperforms CL, a minor performance gap remains compared to PACFL[†]. Additional results and cluster visualizations are provided in the supplementary material.

Ablation Study. We conduct ablation experiments to evaluate the proposed clustering strategy (Tab. 3). In Tab. 3a, we replace our soft clustering with the hard clustering [9] used in PACFL[†], resulting in a hard clustering setting under the same threshold τ . Under the same setup, soft clustering performs better than hard clustering. This suggests that data distributions in robotic environments are continuous and overlapping, making flexible clustering more suitable than hard partitioning.

Table 3: Ablation study. All results are reported using the $Abs\ Rel \downarrow$ metric based on NeWCRFs [44] under the HPE scenario. (a) compares hard and soft clustering under the same threshold τ , (b) evaluates sensitivity to overlap ratios, and (c) reports results across train-test domain splits based on ground-truth labels.

(a) Hard vs. Soft clustering			(b) Overlap sensitivity									
Cluster type	Hard	Soft	Overlap (%)									
$Abs\ Rel \downarrow$	0.264	0.249	$Abs\ Rel \downarrow$	0	10	20	30	40	50	100	FedAvg	
				0.335	0.253	0.249	0.257	0.295	0.300	0.365	0.366	
(c) Domain heterogeneity												
Type	Method	Train set	Test set									
			UGV	UAV	Legged	Forest	Urban-day	Urban-night	Indoor	All		
All	CL	All	0.122	0.158	0.261	0.091	0.112	0.280	0.284	0.167		
	FedAvg	All	0.166	0.372	0.700	0.298	0.358	0.356	0.659	0.366		
	PACFL [†]	All	0.174	<u>0.305</u>	0.580	<u>0.236</u>	0.338	0.414	0.309	<u>0.318</u>		
	Ours	All	<u>0.148</u>	0.262	<u>0.411</u>	0.203	0.185	0.420	0.275	0.249		
Plat.	FedAvg	UGV	0.139	0.514	0.996	0.493	0.366	<u>0.318</u>	0.663	0.471		
	FedAvg	UAV	0.295	0.640	1.358	0.790	0.549	0.471	0.610	0.662		
	FedAvg	Legged	0.458	0.645	0.265	0.407	0.413	0.483	0.782	0.484		
Env.	FedAvg	Forest	0.269	0.383	0.530	0.246	0.351	0.381	0.764	0.370		
	FedAvg	Urban-day	0.191	0.432	0.696	0.415	<u>0.292</u>	0.301	0.680	0.394		
	FedAvg	Urban-night	0.204	0.502	0.943	0.527	0.393	0.330	0.657	0.481		
	FedAvg	Indoor	0.709	0.823	2.012	1.297	1.237	0.764	<u>0.279</u>	1.069		

(a) Hard vs. Soft clustering			(b) Overlap sensitivity									
Cluster type	Hard	Soft	Overlap (%)									
$Abs\ Rel \downarrow$	0.264	0.249	$Abs\ Rel \downarrow$	0	10	20	30	40	50	100	FedAvg	
				0.335	0.253	0.249	0.257	0.295	0.300	0.365	0.366	

(c) Domain heterogeneity												
Type	Method	Train set	Test set									
			UGV	UAV	Legged	Forest	Urban-day	Urban-night	Indoor	All		
All	CL	All	0.122	0.158	0.261	0.091	0.112	0.280	0.284	0.167		
	FedAvg	All	0.166	0.372	0.700	0.298	0.358	0.356	0.659	0.366		
	PACFL [†]	All	0.174	<u>0.305</u>	0.580	<u>0.236</u>	0.338	0.414	0.309	<u>0.318</u>		
	Ours	All	<u>0.148</u>	0.262	<u>0.411</u>	0.203	0.185	0.420	0.275	0.249		
Plat.	FedAvg	UGV	0.139	0.514	0.996	0.493	0.366	<u>0.318</u>	0.663	0.471		
	FedAvg	UAV	0.295	0.640	1.358	0.790	0.549	0.471	0.610	0.662		
	FedAvg	Legged	0.458	0.645	0.265	0.407	0.413	0.483	0.782	0.484		
Env.	FedAvg	Forest	0.269	0.383	0.530	0.246	0.351	0.381	0.764	0.370		
	FedAvg	Urban-day	0.191	0.432	0.696	0.415	<u>0.292</u>	0.301	0.680	0.394		
	FedAvg	Urban-night	0.204	0.502	0.943	0.527	0.393	0.330	0.657	0.481		
	FedAvg	Indoor	0.709	0.823	2.012	1.297	1.237	0.764	<u>0.279</u>	1.069		

Heterogeneous Plat. & Env. (HPE)

Bi-Modal Range (BMR)

Fig. 6: Cluster map and convergence graph. Results for NeWCRFs [44] in the HPE and BMR scenarios. Colors indicate cluster assignments, with each column representing a client. The convergence graphs report $RMSE_{log}$ over communication rounds.

Tab. 3b reports a sensitivity study on the overlap ratio in soft clustering, defined as the fraction of clients assigned to multiple clusters. As shown in the table, a small overlap limits knowledge sharing among related domains, whereas a large overlap increases intra-cluster heterogeneity. In the extreme case of 100% overlap, all clients collapse into a single cluster, making the setting equivalent to FedAvg. As a result, we adopt an overlap ratio of 20%, which yields the best performance in our experiments.

Tab. 3c compares our approach with clustering based on ground-truth labels, where clients are grouped by platform or environment and trained separately before evaluation on the same test set. Although grouping with known labels represents a well defined cluster, unstable performance is observed. This indicates that grouping clients solely by known labels does not necessarily yield optimal performance. In contrast, FEDEPTH performs distribution-aware clustering that captures finer-grained similarities between clients, leading to consistently better results. Additional results and cluster visualizations are provided in the supplementary material.

5 Conclusion

We have proposed FEDEPTH, a clustered federated learning (CFL) framework that addresses the non-IID challenges of depth estimation for robots in the real world. By utilizing soft cluster assignments, FEDEPTH alleviates the performance degradation of FL that is commonly observed under robot-driven heterogeneity. To enable systematic analysis, we design and construct two realistic and previously unexplored robot heterogeneity scenarios that highlight the impact of platform, environment, and depth range variations. These scenarios, together with our framework, serve as a practical benchmark and foundation for applying FL in diverse real-world robotic systems. We believe that FEDEPTH represents an important step toward scalable perception across heterogeneous robots, bringing FL closer to practical deployment.

Acknowledgements

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2020-II201336, Artificial Intelligence Graduate School Program (UNIST); No.RS-2025-25442824, AI Star Fellowship Program (UNIST); No.RS-2022-II220907, Development of AI Bots Collaboration Platform and Self-organizing), and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.RS-2024-00457065).

References

1. Acar, D.A.E., Zhao, Y., Navarro, R.M., Mattina, M., Whatmough, P.N., Saligrama, V.: Federated learning based on dynamic regularization. In: Int. Conf. Learn. Represent. (2021)
2. Arandjelovic, R., Gronát, P., Torii, A., Pajdla, T., Sivic, J.: NetVLAD: CNN architecture for weakly supervised place recognition. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 5297–5307 (2016). <https://doi.org/10.1109/CVPR.2016.572>
3. Bhat, S.F., Alhashim, I., Wonka, P.: AdaBins: Depth estimation using adaptive bins. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 4009–4018 (2021). <https://doi.org/10.1109/CVPR46437.2021.00400>

4. Bhat, S.F., Alhashim, I., Wonka, P.: LocalBins: Improving depth estimation by learning local distributions. In: Eur. Conf. Comput. Vis. pp. 480–496 (2022). https://doi.org/10.1007/978-3-031-19769-7_28
5. Bhat, S.F., Birkel, R., Wofk, D., Wonka, P., Müller, M.: ZoeDepth: Zero-shot transfer by combining relative and metric depth. arXiv preprint arXiv:2302.12288 (2023)
6. Cao, Y., Wu, Z., Shen, C.: Estimating depth from monocular images as classification using deep fully convolutional residual networks. IEEE Trans. Circuits Syst. Video Technol. **28**(11), 3174–3182 (2018). <https://doi.org/10.1109/TCSVT.2017.2740321>
7. Chaney, K., Cladera, F., Wang, Z., Bisulco, A., Hsieh, M.A., Korpela, C., Kumar, V., Taylor, C.J., Daniilidis, K.: M3ED: Multi-robot, multi-sensor, multi-environment event dataset. In: IEEE Conf. Comput. Vis. Pattern Recog. Worksh. pp. 4016–4023 (2023). <https://doi.org/10.1109/CVPRW59228.2023.00419>
8. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Eur. Conf. Comput. Vis. pp. 833–851 (2018). https://doi.org/10.1007/978-3-030-01234-2_49
9. Day, W.H.E., Edelsbrunner, H.: Efficient algorithms for agglomerative hierarchical clustering methods. J. Classif. **1**, 7–24 (1984). <https://doi.org/10.1007/BF01890115>
10. Deng, Y., Kamani, M.M., Mahdavi, M.: Adaptive personalized federated learning. arXiv preprint arXiv:2003.13461 (2020)
11. Dinh, C.T., Tran, N.H., Nguyen, T.D.: Personalized federated learning with Moreau envelopes. In: Adv. Neural Inform. Process. Syst. vol. 33, pp. 21394–21405 (2020)
12. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: Adv. Neural Inform. Process. Syst. pp. 2366–2374 (2014)
13. Gehrig, M., Scaramuzza, D.: Recurrent vision transformers for object detection with event cameras. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 13884–13893 (2023). <https://doi.org/10.1109/CVPR52729.2023.01334>
14. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The KITTI dataset. Int. J. Robot. Res. **32**(11), 1231–1237 (2013). <https://doi.org/10.1177/0278364913491297>
15. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 3354–3361 (2012). <https://doi.org/10.1109/CVPR.2012.6248074>
16. Ghosh, A., Chung, J., Yin, D., Ramchandran, K.: An efficient framework for clustered federated learning. In: Adv. Neural Inform. Process. Syst. vol. 33, pp. 19586–19597 (2020)
17. Guizilini, V., Ambrus, R., Pillai, S., Raventos, A., Gaidon, A.: 3D packing for self-supervised monocular depth estimation. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 2485–2494 (2020). <https://doi.org/10.1109/CVPR42600.2020.00256>
18. Guo, Y., Tang, X., Lin, T.: Enhancing clustered federated learning: Integration of strategies and improved methodologies. In: Int. Conf. Learn. Represent. (2025)
19. Islam, M.S., Javaherian, S., Xu, F., Yuan, X., Chen, L., Tzeng, N.: FedClust: Tackling data heterogeneity in federated learning through weight-driven client clustering. In: Int. Conf. Parallel Process. pp. 474–483 (2024). <https://doi.org/10.1145/3673038.3673151>
20. Jeffreys, H.: The theory of probability. Oxford University Press, 3 edn. (1998)

21. Karimireddy, S.P., Kale, S., Mohri, M., Reddi, S.J., Stich, S.U., Suresh, A.T.: SCAFFOLD: Stochastic controlled averaging for federated learning. In: *Int. Conf. Mach. Learn. Proceedings of Machine Learning Research*, vol. 119, pp. 5132–5143. PMLR (2020)
22. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W., Dollár, P., Girshick, R.B.: Segment Anything. In: *Int. Conf. Comput. Vis.* pp. 4015–4026 (2023). <https://doi.org/10.1109/ICCV51070.2023.00371>
23. Kou, W., Zhu, G., Cheng, B., Wang, S., Tang, M., Wu, Y.: FedEMA: Federated exponential moving averaging with negative entropy regularizer in autonomous driving. In: *IEEE/RSJ Int. Conf. Intell. Robots Syst.* pp. 10190–10197 (2025). <https://doi.org/10.1109/IRoS60139.2025.11246711>
24. Krizhevsky, A.: Learning multiple layers of features from tiny images. Tech. rep., University of Toronto (2009)
25. Laina, I., Rupperecht, C., Belagiannis, V., Tombari, F., Navab, N.: Deeper depth prediction with fully convolutional residual networks. In: *Int. Conf. 3D Vis.* pp. 239–248 (2016). <https://doi.org/10.1109/3DV.2016.32>
26. Lee, J., Cho, G., Park, J., Kim, K., Lee, S., Kim, J.H., Jeong, S.G., Joo, K.: SlaBins: Fisheye depth estimation using slanted bins on road environments. In: *Int. Conf. Comput. Vis.* pp. 8765–8774 (2023)
27. Li, T., Sahu, A.K., Zaheer, M., Sanjabi, M., Talwalkar, A., Smith, V.: Federated optimization in heterogeneous networks. In: *Proc. Mach. Learn. Syst.* vol. 2, pp. 429–450 (2020)
28. Li, X., Huang, K., Yang, W., Wang, S., Zhang, Z.: On the convergence of FedAvg on non-IID data. In: *Int. Conf. Learn. Represent.* (2020)
29. Liu, Y., Huang, A., Luo, Y., Huang, H., Liu, Y., Chen, Y., Feng, L., Chen, T., Yu, H., Yang, Q.: FedVision: An online visual object detection platform powered by federated learning. In: *AAAI Conf. Artif. Intell.* vol. 34, pp. 13172–13179 (2020). <https://doi.org/10.1609/AAAI.V34I08.7021>
30. McInnes, L., Healy, J., Melville, J.: UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426* (2018)
31. McMahan, B., Moore, E., Ramage, D., Hampson, S., Agüera y Arcas, B.: Communication-efficient learning of deep networks from decentralized data. In: *Int. Conf. Artif. Intell. Stat. Proceedings of Machine Learning Research*, vol. 54, pp. 1273–1282. PMLR (2017)
32. Misra, I., Girdhar, R., Joulin, A.: An end-to-end transformer model for 3D object detection. In: *Int. Conf. Comput. Vis.* pp. 2906–2917 (2021). <https://doi.org/10.1109/ICCV48922.2021.00290>
33. Poggi, M., Tosi, F.: Federated online adaptation for deep stereo. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 20165–20175 (2024). <https://doi.org/10.1109/CVPR52733.2024.01906>
34. Radenović, F., Tolias, G., Chum, O.: Fine-tuning CNN image retrieval with no human annotation. *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(7), 1655–1668 (2019). <https://doi.org/10.1109/TPAMI.2018.2846566>
35. Ranftl, R., Bochkovskiy, A., Koltun, V.: Vision transformers for dense prediction. In: *Int. Conf. Comput. Vis.* pp. 12179–12188 (2021). <https://doi.org/10.1109/ICCV48922.2021.01196>
36. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: *Adv. Neural Inform. Process. Syst.* pp. 91–99 (2015)

37. de S. Soares, E.F., Campos, C.A.V.: Self-supervised learning of monocular depth estimators in autonomous vehicles with federated learning. *Eng. Appl. Artif. Intell.* **151**, 110572 (2025). <https://doi.org/10.1016/j.engappai.2025.110572>
38. Sattler, F., Müller, K.R., Samek, W.: Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE Trans. Neural Netw. Learn. Syst.* **32**(8), 3710–3722 (2021). <https://doi.org/10.1109/TNNLS.2020.3015958>
39. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from RGBD images. In: *Eur. Conf. Comput. Vis.* pp. 746–760 (2012). https://doi.org/10.1007/978-3-642-33715-4_54
40. Vahidian, S., Morafah, M., Wang, W., Kungurtsev, V., Chen, C., Shah, M., Lin, B.: Efficient distribution similarity identification in clustered federated learning via principal angles between client data subspaces. In: *AAAI Conf. Artif. Intell.* vol. 37, pp. 10043–10052 (2023). <https://doi.org/10.1609/aaai.v37i8.26197>
41. Wang, K., Yan, Z., Fan, J., Zhu, W., Li, X., Li, J., Yang, J.: DCDepth: Progressive monocular depth estimation in discrete cosine domain. In: *Adv. Neural Inform. Process. Syst.* vol. 37, pp. 64629–64648 (2024)
42. Yang, L., Kang, B., Huang, Z., Xu, X., Feng, J., Zhao, H.: Depth Anything: Unleashing the power of large-scale unlabeled data. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 10371–10381 (2024). <https://doi.org/10.1109/CVPR52733.2024.00987>
43. Yang, L., Kang, B., Huang, Z., Zhao, Z., Xu, X., Feng, J., Zhao, H.: Depth Anything V2. In: *Adv. Neural Inform. Process. Syst.* vol. 37, pp. 21875–21911 (2024)
44. Yuan, W., Gu, X., Dai, Z., Zhu, S., Tan, P.: Neural window fully-connected CRFs for monocular depth estimation. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 3916–3925 (2022). <https://doi.org/10.1109/CVPR52688.2022.00389>
45. Zhang, J., Liu, H., Yang, K., Hu, X., Liu, R., Stiefelhagen, R.: CMX: Cross-modal fusion for RGB-X semantic segmentation with transformers. *IEEE Trans. Intell. Transp. Syst.* **24**(12), 14679–14694 (2023). <https://doi.org/10.1109/TITS.2023.3300537>
46. Zhang, Y., Chen, H., Lin, Z., Chen, Z., Zhao, J.: LCFed: An efficient clustered federated learning framework for heterogeneous data. In: *IEEE Int. Conf. Acoust. Speech Signal Process.* pp. 1–5 (2025). <https://doi.org/10.1109/ICASSP49660.2025.10889428>
47. Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., Chandra, V.: Federated learning with non-IID data. *arXiv preprint arXiv:1806.00582* (2018)
48. Zheng, H., Hu, Z., Yang, L., Zheng, M., Xu, A., Wang, B.: FedCALM: Conflict-aware layer-wise mitigation for selective aggregation in deeper personalized federated learning. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 15444–15453 (2025)