A Dataset and Experimental Study towards Visual Terrain Classification with OOD Challenges in Real World

Haotian Zhou*, Jianghuan Xu*, Hongze Li*, Rui Xie*, Huijing Zhao*
*Peking University, Beijing, China

***Abstract*— In terrain classification task, data within the same terrain exhibit a high intra-class diversity, often leading to a common real-world Out-of-Distribution (OOD) problem. Some studies employ rigorous methods to control the similarity between the training and test set distributions, thereby ignoring this real-world OOD problem. To delve into this challenge, we collect a high intra-class diversity terrain classification dataset TCPOSS. We observe a significant performance decline when terrain classification models encounter real-world OOD data. To quantify the severity of real-world OOD, we propose a metric *KLConf*. Experiments show a strong correlation between *KLConf* and the decline of model performance on the test set. The dataset and relative codes will be released at https://github.com/weekgoodday/TCPOSS later.**

I. Introduction

Terrain classification is an important task for autonomous mobile robots to figure out what kind of surface they are on. With this function, robots can plan a path that is as safe and fast as possible [1], [2], [3]. With the development of deep learning [4], visual terrain classification methods achieve near-perfect performance when test scenes are similar to training scenarios [1].

However, when applying the model trained on [1] to classify terrains in campus environments, we observe a drastic decline in performance. Certain categories such as sand terrain are completely unidentifiable by the model. Similar phenomenon is observed in [5][6]. The reason lies that the deep model does not truly learn to distinguish between asphalt, grass and sand terrains; instead, it has learned "shortcut" to distinguish between classes present in the training data [7]. This phenomenon severely limits the applicability of deep models. This kind of problem arises especially when the training set cannot comprehensively cover all data distributions within a class due to intra-class diversity. This real-world Out-of-Distribution (OOD) problem, which commonly occurs in real-world scenarios, has been scarcely discussed within the existing OOD frameworks.

A major characteristic of terrain data is that the same type of terrain may exhibit significant intra-class diversity (see Fig. 1). And it's expensive to collect and label terrain data [2]. Therefore, visual terrain classification model may not traverse all possible data representations of existing categories during training.

To delve into this real-world OOD challenge, we collect a terrain classification dataset of high intra-class diversity, and conduct different training and test set divisions based on the severity of the OOD problem. Then we analyze the model performance when facing OOD data of various intensities.

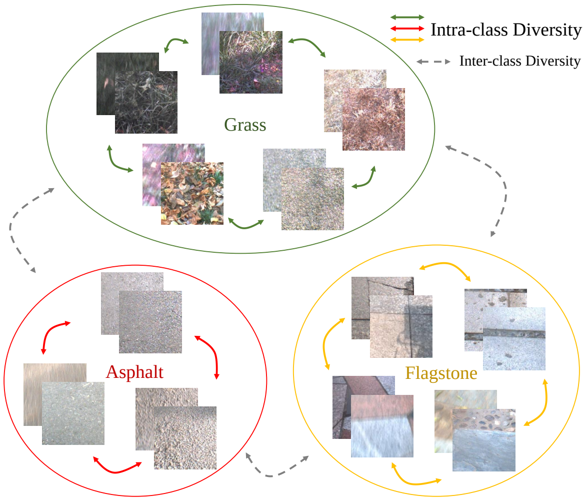

Fig. 1. Intra-class diversity illustration in terrain classification. The green represents grass class, red represents asphalt, yellow presents flagstone.

To figure out whether the model's output can be believed, we apply multiple classic confidence estimation methods to have the terrain classification model output a confidence score alongside its prediction. With confidence evaluation, we propose *KLConf* metric to quantify the disparity between training and deployment scenarios.

The contributions are summarized as follows:

1) A terrain classification dataset named TCPOSS is proposed, which comprises terrain data of 10 categories in Peking University, reflecting rich intra-class diversity. This dataset is divided into four subsets: Entry, Easy, Medium and Hard, with test data reflecting OOD challenges from easy to hard. To the best of our knowledge, this is the first dataset comprising various degrees of OOD problems for terrain classification.
2) Experiments are conducted to analyze the influence of various OOD intensities on the performance of terrain classification model. Experimental results show models with various architectures and configurations all experience performance degradation when facing this OOD problem. Feature space analysis indicates that intra-class diversity is the main reason for this degradation in real-world datasets.
3) A method to quantify the OOD intensity is proposed. The *KLConf* metric evaluates the difference in confidence distribution between training and test set. Experiments show that *KLConf* can effectively quantify the degree of real-world OOD and predict the performance of classification model on test set without ground truth.

The rest of this paper is organized as follows. Section II is the related works and datasets about terrain classification, OOD problem. Section III introduces our dataset TCPOSS. Section IV shows the performance of terrain classification model. To quantify the OOD problem, Section V compares classic confidence estimation methods and proposes *KLConf*. Finally, Section VI gives the conclusions and future works.

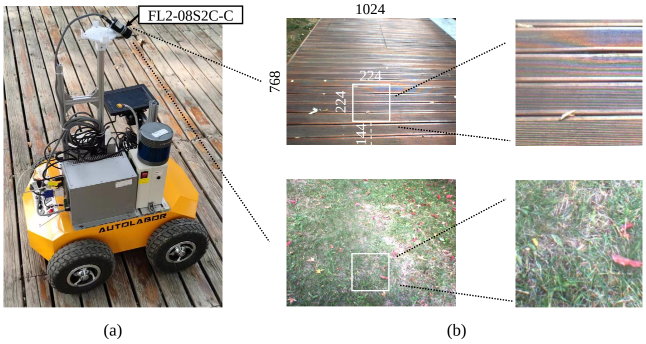

Fig. 2. (a) Platform used for data collection; (b) Data process pipeline.

II. Related Works

A. Terrain Classification

From the perspective of sensor inputs, the solution of terrain classification can be classified into two main categories: exteroceptive-based and proprioceptive-based [8]. Camera is the most popular exteroceptive sensor [2]. There are also some researches using input signals like spectral data [9] and acoustic data [10]. Every kind of signal has its characteristics. The proprioceptive signal won't be interfered by light conditions, but it's a contact-based signal and is susceptible to robot body self-vibration [2]. This article mainly focuses on image signals. Compared with proprioceptive signals, images have rich textures and can be acquired before contact.

Some terrain classification studies use terrain images similar to our datasets [1], [9], [11], [12], [13]. But only in [1] and [9], the terrain images are publicly available. The training and test sets of both datasets have already been sampled from sequences and exhibit similar morphologies. The OOD problem in current datasets is not obvious and the intra-class diversity is ignored, which deviates from the real-world scenarios. To the best of our knowledge, our dataset TCPOSS is the first dataset comprising various degrees of OOD problems for visual terrain classification.

B. OOD Problem

The Out-of-Distribution (OOD) problem refers to the scenario where a model encounters test data that does not belong to the potential distribution of the training set [14]. In such cases, the model should not be trusted or used for inference. Due to the unknown potential distribution of the training set, there is no unified definition of OOD data. Faced with OOD problem, there are two basic pipelines to address it: OOD detection [15] and OOD generalization [16]. Not all OOD data can be generalized by models, but with OOD detection we can at least identify whether the data exceeds the model's applicability, making the system safer and more robust [17]. Among OOD detection methods, those based on model uncertainty or confidence fully leverage the intrinsic characteristics of the model, requiring no specification of OOD data or excessive model modifications [18].

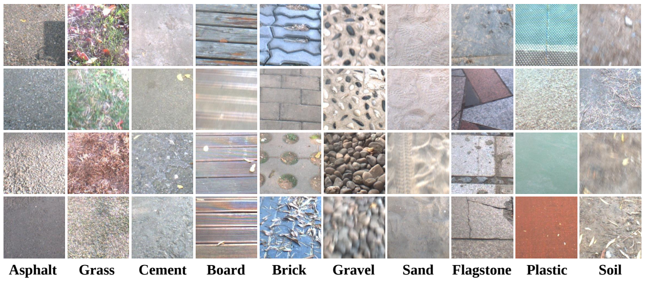

Fig. 3. Sample images of 10 terrain classes of TCPOSS.

Public datasets in OOD detection primarily use image classification datasets sourced from different origins [19], [20]. Some researches define certain datasets as ground truth for OOD [21], [22], while others generate OOD images through generative models [23]. However, these benchmarks do not generally represent realistic distribution shifts, such as train/test splits that are likely to occur in real-world deployments [24], [25].

III. Datasets

A. Real-world Terrain Data Collection

This research collects a visual terrain classification dataset, named TCPOSS. The data are collected in Peking University, where terrains with diverse morphologies are abundant throughout the campus. The data are collected in autumn, and some terrain surfaces are covered with fallen leaves, causing interference to visual classification. We collect data during the day and in the evening, the change in light also introduces disturbance to visual classification. Our UGV, shown in Fig. 2, has a FL2-08S2C-C camera. With an original resolution of 768*1024, we crop a 224*224 image patch close to the ground as the data input. Through manual control, the speed of the robot when collecting terrain data is between 0.5m/s and 1m/s, simulating real-world robot movement states.

As shown in Fig. 3, the collected data are categorized into 10 classes based on terrain surface, including: asphalt, grass, cement, brick, board, gravel, sand, flagstone, plastic, soil. The data are sequential, and every 1 second includes 15 image frames. We use the first frame per second as input.

TABLE I

Datasize and sequences of 10 classes in our dataset.

	Asphalt	Grass	Cement	Brick	Board	Gravel	Sand	Flagstone	Plastic	Soil	Total
Datasize	1299	1158	818	138	682	160	231	1322	1169	225	7202
Sequences	9	13	9	3	9	7	3	19	14	4	90

TABLE II

Datasize and Features of each category in different Dataset Divisions

Dataset	Datasize of each category (Train/Test)											Relation between Train/Test	Feature
	As	Gr	Ce	Bo	Br	Gr	Sa	Fl	Pl	So	Total		
Entry	897/ 402	805/ 353	708/ 289	97/ 41	489/ 193	111/ 49	154/ 77	930/ 392	818/ 351	157/ 68	5166/ 2215	same sequence similar appearance	ID
Easy	1103/ 196	957/ 201	625/ 193	79/ 59	545/ 137	87/ 73	154/ 77	1117/ 205	905/ 264	134/ 91	5706/ 1496	different sequences in same area relatively similar appearance	Low OOD
Medium	894/ 405	752/ 406	532/ 286	79/ 59	528/ 154	87/ 73	154/ 77	1040/ 282	905/ 264	134/ 91	5105/ 2097	different sequences in same area relatively different appearance	Medium OOD
Hard	894/ 405	769/ 389	594/ 224	79/ 59	467/ 215	116/ 44	154/ 77	958/ 364	789/ 380	134/ 91	4954/ 2248	sequences in different areas different appearance	High OOD

As shown in Tab. I, after selecting valid segments, there are a total of 7202 valid image frames, in 90 independent sequences.

B. OOD Dataset Generation

As shown in Tab. II, according to the difference between training and test set, we divide the dataset into four levels: Entry, Easy, Medium and Hard. Entry randomly selects 30% of the images as the test set with remaining 70% as the training set. Thus, all categories in training set and test set are from the same sequence with similar appearance, which can be considered within the same distribution. For Easy, compared with training set, all categories in test set are from different sequences in the same area with relatively similar appearance. For Medium, compared with training set, all categories in test set are from different sequences in the same area, but the appearance of some categories is relatively different from training set. For hard, some categories are from sequences in different areas with different appearance.

Taking the grass category as an example, Fig. 4 specifically illustrates our criteria for dividing training and test set. For Hard, the data from Area A in the evening along with all the data from Area C are used as the test set, while the rest are used as the training set. For Medium, the data from Area A in the evening, along with part of the data from Area C and D, are used as the test set, while the rest are used as the training set. For Easy, one sequence from Area A during the day and one sequence from Area B are used as the test set.

In summary, we don't artificially define which data is OOD, while there exist various degrees of OOD challenges in different divisions. By rationally dividing the training and test sets, we aim to simulate the common OOD problem caused by intra-class diversity frequently encountered in real-world environments.

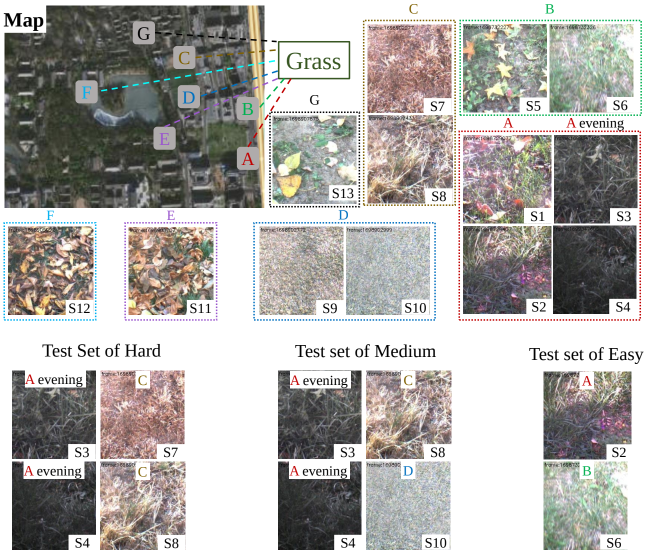

Fig. 4. Illustration of dataset division (taking grass class as an example). The grass is all around the campus and exhibits diverse morphologies. The Hard division uses S3, S4, S7 and S8 as test set, while others as training set, which encounters severe OOD in test set. Similarly, the Medium and Easy includes illustrated sequences in test set, while others as training set.

IV. Terrain Classification on Real-world OOD Dataset

In this section, we evaluate the terrain classification model on both the public dataset [1] and our dataset. Through model performance and feature space of every category, we analyze the reasons for model performance degradation and compare the characteristics of each category across datasets. We use "Dataset [1]" to denote the public dataset in [1].

A. Terrain Classification Model

The deep classification network adopts the architecture described in [1], which consists of two dense layers after the

TABLE III

Model configurations and corresponding performance on various datasets

Model configuration			Performance on test set				
model in [1]	no freezing parameters & data augmentation	more training epochs & learning rate decay	Dataset [1]	Entry	Easy	Medium	Hard
✓			99.7%(±0.2%)	92.7%(±0.7%)	77.2%(±2.9%)	57.1%(±2.0%)	45.7%(±1.7%)
✓	✓		99.9%(±0.1%)	94.6%(±2.2%)	78.6%(±5.7%)	60.7%(±5.7%)	45.4%(±6.3%)
✓	✓	✓	>99.95%	98.9%(±0.2%)	88.1%(±3.2%)	68.2%(±5.5%)	45.8%(±2.6%)

TABLE IV

DIFFERENT BACKBONES AND CORRESPONDING PERFORMANCE

Backbone	Performance on test set				
	Dataset [1]	Entry	Easy	Medium	Hard
Mobilenet	>99.95%	98.9%	88.1%	68.2%	45.8%
Densenet	>99.95%	99.1%	88.8%	70.1%	49.2%
Resnet	>99.95%	99.2%	87.6%	67.2%	48.1%

backbone of MobilenetV2 [26]. This architecture produces a logit vector of dimensions corresponding to the number of classes, which is optimized using cross-entropy loss with one-hot label. In [1], the parameters of the MobilenetV2 backbone are frozen, utilizing the pre-trained parameters on ImageNet, only training the subsequent classifier.

Building upon this foundation, we enhance the model by removing parameter freezing and introducing additional data augmentation techniques, including color normalization, horizontal/vertical flipping, and random angle rotation. We extend the training epochs to 40 and incorporate exponential learning rate decay. As shown in Tab. III, the performance after applying these configurations on various datasets has been improved compared to the origin. Based on the optimal configurations, we switch to different feature extraction backbones, such as Densenet121 [27] and Resnet50 [28] as shown in Tab. IV. The results indicate that the performance gap introduced by different backbones is not substantial. Dataset selection emerges as the primary factor influencing model performance.

Without loss of generality, this study focuses on the Mobilenet backbone and the optimal model configurations. The following analysis can reflect the common characteristics of deep visual classification models.

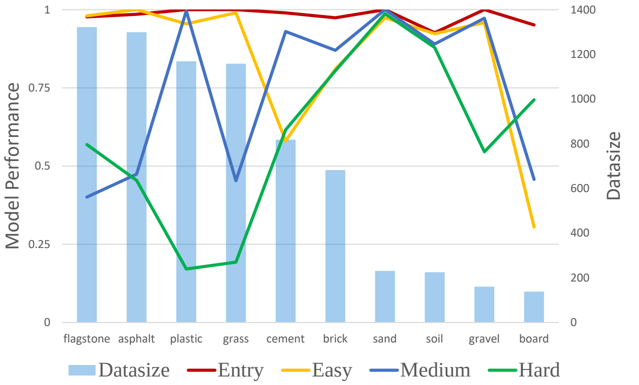

Fig. 5. Performance on different categories. Lines of different colors represent the recall of each category on different datasets, while the bars represent the data size of each category.

B. Terrain Classification Results

As shown in Tab. III and IV, our datasets are relatively more challenging. When the test set is in-distribution, the model achieves a near-perfect performance. However, as the model encounters strong OOD problems, there is a significant decline in performance. The magnitude of performance degradation is independent of architectures of deep models, indicating a common issue faced by deep models.

Besides the overall accuracy, different categories face various degrees of OOD across different datasets as shown in Fig. 5. Some categories, such as sand and soil, exhibit similar performance across different datasets as their visual morphologies in campus tend to be uniform. While categories with high intra-class diversity, such as grass and flagstone, show significant performance differences across different datasets.

C. Real-world OOD Problem Analysis

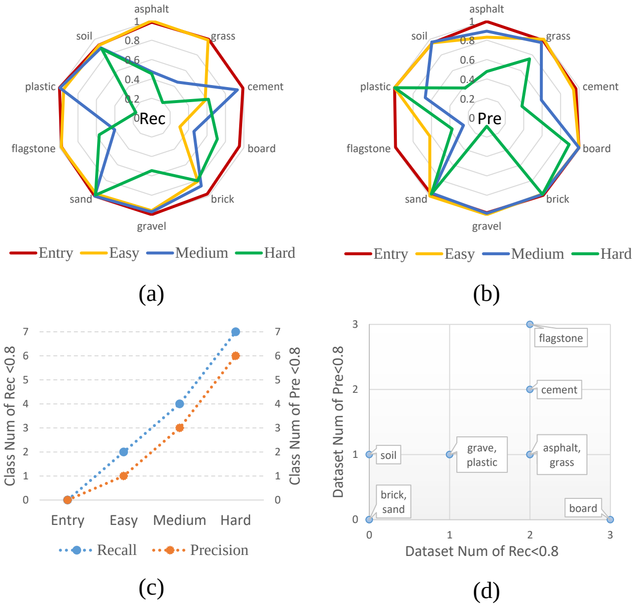

Fig. 6. Categorical analysis of OOD intensity on 4 datasets. (a) Radar chart illustrating the recall of each category across 4 datasets. (b) Radar chart illustrating precision. (c) The trend in the total number of categories with recall/precision less than 0.8 across 4 datasets. (d) Performance of each category across 4 datasets. The x-axis represents the number of datasets on which the recall is less than 0.8, and the y-axis represents precision.

We begin by further analyzing the recall and precision of each category. As shown in Fig. 6, the areas enclosed by polygons of different colors in (a) and (b) indicate the mean recall and precision of 10 categories across 4 datasets. (c) shows the total number of categories with precision and recall below 0.8. The performance degradation reflects the OOD degree of datasets. (d) presents the performance of each category based on recall and precision. Some additional conclusions can be drawn from the categorical analysis. For instance, categories like grass, asphalt, cement, and flagstone in our dataset encounter strong OOD problems in Medium and Hard. The board category exhibits low recall and high precision, indicating that the model is less likely to output board class, which may be attributed to the small data size.

Besides the accuracy, we observe the model's feature space. Sand, plastic, grass and flagstone are selected as representative categories. As shown in Fig. 7, the dashed circles illustrate the potential training distribution of these categories in the feature space after dimension reduction. Overall, data of the same category in Easy exhibit a better correspondence between training and test set compared to the Medium and Hard. The classes with low recall share a common characteristic: the data clusters in the test set are

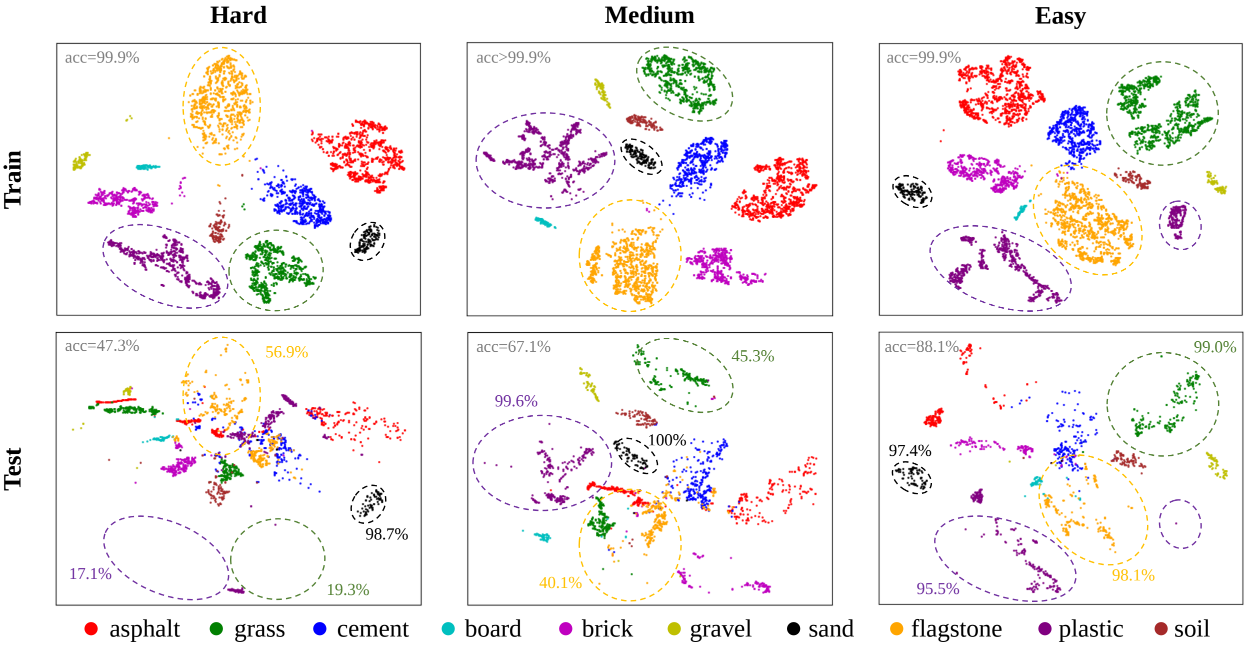

Fig. 7. Feature space visualization on Hard, Medium and Easy. After passing through the feature extraction network, the 1280-dimensional features of the training and test sets are jointly subjected to T-SNE, then plot separately on two graphs. The first row represents the feature space of the training set, while the second row represents the test set. Different colors indicate the different labels of each data point. The annotated numbers represent the recall of specific category.

not in the distribution of the training set. This indicates that the decline in performance is caused by training data failing to comprehensively cover the entire potential space due to intra-class diversity. Moreover, on the test set of Hard, the grass points fall within the distributions of the gravel and soil categories, corresponding to the fact that inputs of the grass class are misclassified to gravel and soil. This suggests that the model learns feature extraction methods through the training data, but features extracted by the learned methods may not be comprehensive enough to distinguish test data, leading to confusion between categories.

In reality, the most direct way to address the OOD problem caused by intra-class diversity is to collect as comprehensive training data as possible, enabling the model to extract sufficient features for classification. However, it is almost impossible to exhaustively capture the intra-class diversity in the real world. Therefore, the first step in addressing this OOD problem is to quantify the OOD intensity on test samples without ground truth. Then we can select OOD samples and incorporate them into training.

V. Quantification of Real-world OOD Problem via Confidence Evaluation

To address the decline in model performance caused by OOD problems, this study explores whether the model can detect the presence of OOD in testing and measure the severity of OOD. In the methods of OOD detection, confidence-based methods require only simple modifications to the model and are applicable to classification models. This section compares five classical confidence estimation methods. Experimental results demonstrate that existing confidence methods exhibit performance degradation when confronted with strong OOD scenarios, but they can effectively reflect the differences between the test distribution and the training distribution. Based on confidence methods, We propose *KLConf* metric to quantify the OOD intensity of test set.

A. Confidence Methods

This section compares five classical confidence estimation methods: Softmax Confidence (SM) [19], MC Dropout (MC) [29], Ensemble (EM) [30], Mahalanobis Distance (MD) [31], Evidential Deep Learning (EDL) [32].

Specifically, in a classification task, we denote the input of deep model as $\boldsymbol{x}$, and the last layer output of class c as $f_c(\boldsymbol{x})$. Generally, the vector after Softmax layer is regarded as the probability of model prediction $\boldsymbol{p}(\boldsymbol{x})$. It's a discrete vector with the same dimension as the number of classes C:

$$p_c(\boldsymbol{x}) = \frac{\exp(f_c(\boldsymbol{x}))}{\sum_{i=1}^{C}\exp(f_i(\boldsymbol{x}))} \tag{1}$$

where $p_c(\boldsymbol{x})$ is the c^{th} component of $\boldsymbol{p}(\boldsymbol{x})$, representing the predicted probability of class c.

The Softmax Confidence is the maximum component of the predicted probability $\boldsymbol{p}(\boldsymbol{x})$:

$$SM_{conf}(\boldsymbol{x}) = \max_c p_c(\boldsymbol{x}) \tag{2}$$

1) Calibration-based: As Softmax confidence is always overconfident [33], temperature scaling is applied to the Softmax confidence with a temperature parameter T [34]:

$$Temp_{conf}(\boldsymbol{x}) = \max_c \frac{\exp(f_c(\boldsymbol{x})/T)}{\sum_{i=1}^{C}\exp(f_i(\boldsymbol{x})/T)} \tag{3}$$

Subsequently, the confidence value is calibrated downward, while maintaining the relative order of confidence among the samples.

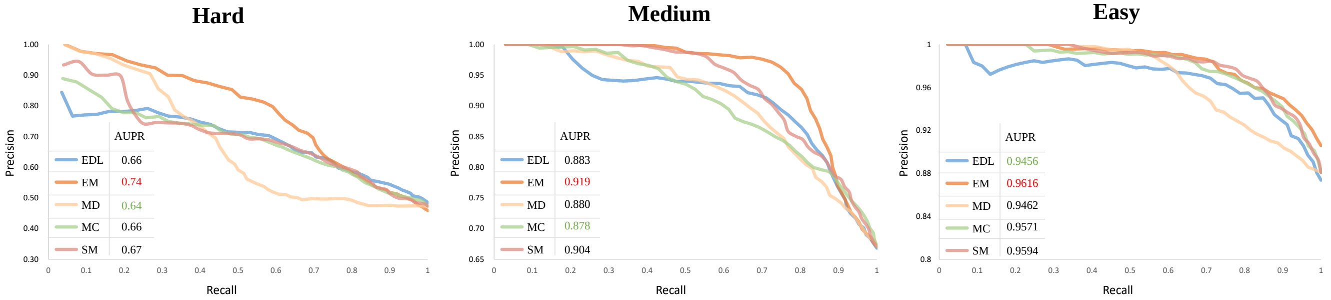

Fig. 8. PR Curve on Hard, Medium, Easy. AUPR (Area Under Precision-Recall curve) values of 5 confidence methods are annotated in the bottom left corner of the graph. Values in red mean the maximum AUPR, while green minimum.

2) Ensemble-based: Given an ensemble $\{\mathcal{M}\}_{m=1}^{M}$, the total uncertainty can be modeled as aleatoric uncertainty and epistemic uncertainty [35], which can be given by:

$$\begin{aligned} au(\boldsymbol{x}) &= \mathbb{E}_{\mathcal{M}}[H(\boldsymbol{p}(\boldsymbol{x},\mathcal{M}))] \\ eu(\boldsymbol{x}) &= H(\mathbb{E}_{\mathcal{M}}[\boldsymbol{p}(\boldsymbol{x},\mathcal{M})]) - au(\boldsymbol{x}) \end{aligned} \tag{4}$$

For Monte-Carlo Dropout, $\boldsymbol{p}(\boldsymbol{x},\mathcal{M})$ is acquired by multiple forward passes with Dropout layers activated. For Ensemble, M models are trained to get $\boldsymbol{p}(\boldsymbol{x},\mathcal{M})$. The negative of uncertainty can be viewed as confidence.

3) Distance-based: The distance between the test sample and class center of training samples $\boldsymbol{\mu}_c$ in feature space can be used to measure confidence. Mahalanobis Distance [31] calculates covariance matrix $\boldsymbol{\Sigma}$ of training samples in the feature space, and uses the negative of distance as the confidence with predicted class $\hat{y}$:

$$MD_{conf}(\boldsymbol{x}) = -(\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{\mu}_{\hat{y}})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{\mu}_{\hat{y}}) \tag{5}$$

4) EDL-based: Based on the Dempster-Shafer Theory (DST) [36] and the subjective logic (SL) [37], evidential deep learning (EDL) [32] is developed to learn a better metric of confidence following a prior Dirichlet distribution. Given a training sample $\boldsymbol{x}^{(i)}$, the loss function is thus defined:

$$\mathcal{L}_{EDL}^{(i)}(\boldsymbol{y}^{(i)}, \boldsymbol{e}^{(i)}) = \sum_{c=1}^{C} y_c^{(i)} \left(\log S^{(i)} - \log(e_c^{(i)} + 1) \right) \tag{6}$$

where $\boldsymbol{e}^{(i)} \in \mathbb{R}_+^C$ is the learning evidence. S is the strength of a Dirichlet distribution $\text{Dir}(\boldsymbol{p}|\boldsymbol{\alpha})$ and is defined as $S = \sum_{c=1}^{C} \alpha_c$. α_c can be calculated as $\alpha_c = e_c + 1$.

In EDL, the confidence of predicted class $\hat{y}$ is given by:

$$EDL_{conf}(\boldsymbol{x}) = \alpha_{\hat{y}} / S \tag{7}$$

The detailed implementation of confidence methods is described as follows. For MC, our model incorporates two Dropout layers in the classifier. When testing with MC Dropout, a sample will forward pass five times with Dropout layers activated. For EM, we train five random initialized models with data augmentation. For MD, the covariance matrix $\boldsymbol{\Sigma}$ becomes a singular matrix because of the high-dimensional feature space. We use PCA to reduce the dimension as there is considerable redundancy within the feature space. For EDL, to mitigate the miscalibration linked to the over-fitting of the negative log-likelihood (NLL), we incorporate evidential uncertainty calibration (EUC) loss $\mathcal{L}_{EUC}$ [38]. The total loss function of EDL is:

$$\mathcal{L}_{total} = \mathcal{L}_{EDL} + 0.2 \times \mathcal{L}_{EUC} \tag{8}$$

B. Confidence Evaluation Results

First, we observe the performance of classic confidence estimation methods when facing various degrees of OOD data. We don't have the ground truth of which data is OOD, but we can utilize precision-recall (PR) curves and the area under the PR curve (AUPR) to measure the relationship between confidence scores and correct predictions. We define whether a sample is considered as a correct classification by the model (i.e., whether confidence is greater than a threshold δ) as positive (P) or negative (N), and whether it is truly classified correctly as true (T) or false (F). Then we can plot the PR curve as shown in Fig. 8. Each point on the curve corresponds to a confidence threshold δ. Ideally, misclassified samples should correspond to relatively low confidence values, and correctly classified samples should correspond to high confidence values. In this ideal case,

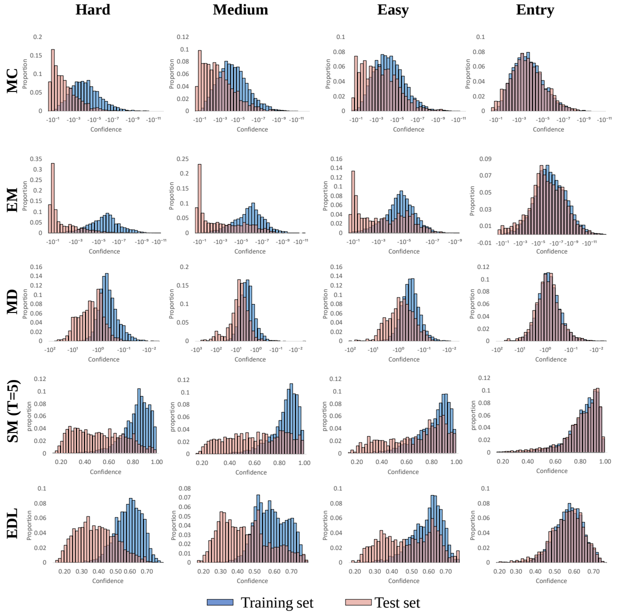

Fig. 9. Confidence distributions on training and test sets. The horizontal axes for MC, EM, and MD are logarithmic, while SM and EDL are uniformly distributed in $[0, 1]$. SM is calibrated with temperature T=5.

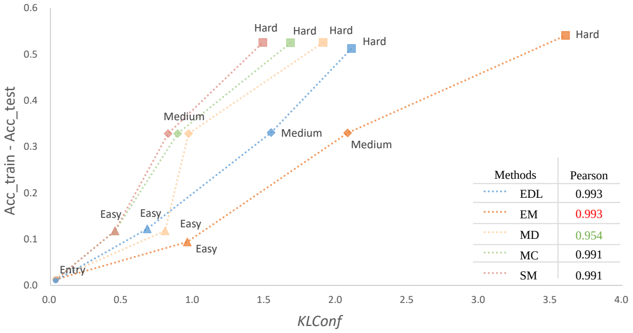

Fig. 10. The strong correlation between *KLConf* and the performance degradation. Different colors represent different confidence evaluation methods, and the Pearson correlation coefficient used to measure this correlation is annotated in the bottom right corner.

the PR curve would tend towards the upper right corner, forming a square with AUPR = 1. It can be observed that on Easy, confidence can effectively distinguish samples with low accuracy. As the severity of OOD increases, there is a larger discrepancy between confidence metrics and the ideal. Compared to other methods, confidence based on EM performs the best. SM, though simple, also exhibits relatively good performance.

Further, we examine the confidence distribution on the training and test sets evaluated by various confidence methods. As shown in Fig. 9, it can be observed that in Entry, which represents ID scenario, the confidence distribution on the test set closely resembles that on the training set. However, when encountering OOD problems, the distribution on test set shifts left compared to the training set. The disparity between the two distributions increases as the severity of the OOD increases. This observation aligns with the intuitive understanding of confidence and demonstrates that existing confidence methods can effectively reflect differences between training and test distributions. Intuitively, on our dataset, confidence measured by EM exhibits the most pronounced difference between training and test sets.

C. *Quantification of Real-world OOD Problem*

Inspired by Fig. 9, we propose the metric *KLConf* by calculating the Kullback-Leibler (KL) divergence between the confidence distribution of training and test set. Subsequently, we compare *KLConf* with the decrease in test set accuracy relative to the training set, as depicted in Fig. 10.

Specifically, we denote $\boldsymbol{c_{tr}}$ as the confidence of training set, $\boldsymbol{c_{te}}$ as confidence of test set, $m_c = min(\boldsymbol{c_{tr}}, \boldsymbol{c_{te}})$, $M_c = max(\boldsymbol{c_{tr}}, \boldsymbol{c_{te}})$. We evenly split $[m_c,\ M_c]$ into N bins for SM and EDL on regular coordinates, while on logarithmic coordinates for EM, MC and MD. Taking SM as an example, we denote $c_{tr}(i)$ as the proportion of confidence falling in $[m_c + (i-1)/N * (M_c - m_c),\ m_c + i/N * (M_c - m_c)]$ for $\boldsymbol{c_{tr}}$. $c_{te}(i)$ likewise. Then, *KLConf* can be calculated as:

$$KLConf = KL(\boldsymbol{c_{te}}||\boldsymbol{c_{tr}}) = \sum_{i=1}^{N} c_{te}(i) log(\frac{c_{te}(i)}{c_{tr}(i)}) \quad (9)$$

On four datasets, we compute the Pearson correlation coefficient between *KLConf* and the decrease in test set accuracy relative to the training set accuracy, as shown in Fig. 10. Typically, an absolute Pearson correlation coefficient greater than 0.8 is considered a strong correlation. Experimental results indicate a strong correlation between *KLConf* and the decline in model performance on the test set.

Furthermore, we investigate whether this strong correlation holds for different categories. We select the EM and EDL methods, which exhibit the best overall performance, and the SM method, which demonstrates relatively good performance with the most convenience. As shown in Tab. V, for each category, we select samples from the training and test sets whose ground truth is that category. The final column of Tab. V calculates the Pearson correlation coefficient for *KLConf* and accuracy decrease across 10 categories and 4 datasets. Experimental results demonstrate that the aforementioned strong correlation persists, especially in categories with large data size. Leveraging this property in practical applications, users can be informed in advance of the model's reliability without the need for labels on the test set, or alternatively, we can select OOD samples and incorporate them into training.

VI. Conclusions and Future Works

In this paper, we provide a high intra-class diversity terrain classification dataset, TCPOSS. In random division, i.e., when data is entirely in-distribution, models are capable of learning and overcoming the intra-class diversity in reality. But when the severity of OOD increases, the performance of deep classification model decreases accordingly. Due to the costs associated with data collection and labeling, training data often fail to comprehensively cover the entire potential space. We propose *KLConf* by calculating the KL divergence between confidence distribution of training and test set. Experiments show that by calculating *KLConf*, it is possible to anticipate the model performance on the test set without requiring ground truth.

The future work can be extended in the following three aspects. For terrain classification tasks, considering the integration of temporal information can yield robust results.

TABLE V
Pearson Correlation Coefficient for EM, EDL and SM in Each Category

	flagstone	asphalt	plastic	grass	cement	brick	sand	soil	gravel	board	total*
Pearson of EM	0.959	0.862	0.961	0.984	0.975	0.906	0.804	0.847	0.996	0.759	0.908
Pearson of EDL	0.910	0.866	0.948	0.991	0.988	0.970	0.798	0.866	0.996	0.888	0.880
Pearson of SM	0.950	0.908	0.969	0.839	0.996	0.955	0.821	0.079	0.996	0.927	0.875
Datasize	1322	1299	1169	1158	818	682	231	225	160	138	7202

Further, terrain traversability with a broader field of view can be explored. For OOD problems, we can design detection methods considering the intra-class diversity. Solutions may draw inspiration from OOD generalization and data generation techniques. For real-world applications, we can explore reasonable output forms beyond predefined categories considering both the downstream tasks and model performance.

References

[1] Yu Chen, Chirag Rastogi, and William R Norris. A cnn based vision-proprioception fusion method for robust ugv terrain classification. *IEEE Robotics and Automation Letters*, 6(4):7965–7972, 2021.

[2] Zhenhua Yu, SM Hadi Sadati, Shehara Perera, Helmut Hauser, Peter RN Childs, and Thrishantha Nanayakkara. Tapered whisker reservoir computing for real-time terrain identification-based navigation. *Scientific Reports*, 13(1):5213, 2023.

[3] Zhen Lu, Mingyi Wang, Shuyang Yu, Yi Wu, You Wang, and Guang Li. Camera-lidar-based terrain multi-type classification using both spatial and histogram features of lidars. In *2023 3rd International Conference on Computer, Control and Robotics (ICCCR)*, pages 298–302. IEEE, 2023.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[5] Damien Teney, Yong Lin, Seong Joon Oh, and Ehsan Abbasnejad. Id and ood performance are sometimes inversely correlated on real-world datasets. *Advances in Neural Information Processing Systems*, 36, 2024.

[6] Timo Milbich, Karsten Roth, Samarth Sinha, Ludwig Schmidt, Marzyeh Ghassemi, and Bjorn Ommer. Characterizing generalization under out-of-distribution shifts in deep metric learning. *Advances in Neural Information Processing Systems*, 34:25006–25018, 2021.

[7] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[8] Paulo Borges, Thierry Peynot, Sisi Liang, Bilal Arain, Matthew Wildie, Melih Minareci, Serge Lichman, Garima Samvedi, Inkyu Sa, Nicolas Hudson, et al. A survey on terrain traversability analysis for autonomous ground vehicles: Methods, sensors, and challenges. *Field Robot*, 2(1):1567–1627, 2022.

[9] Nathaniel Hanson, Michael Shaham, Deniz Erdoğmuş, and Taşkin Padir. Vast: Visual and spectral terrain classification in unstructured multi-class environments. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3956–3963. IEEE, 2022.

[10] Jannik Zürn, Wolfram Burgard, and Abhinav Valada. Self-supervised visual terrain classification from unsupervised acoustic feature learning. *IEEE Transactions on Robotics*, 37(2):466–481, 2020.

[11] Hang Wu, Baozhen Liu, Weihua Su, Zihao Chen, Wenchang Zhang, Xudong Ren, Jinggong Sun, et al. Optimum pipeline for visual terrain classification using improved bag of visual words and fusion methods. *Journal of Sensors*, 2017, 2017.

[12] Song Zeng, Hao Huang, and Zhenyun Shi. Outdoor terrain recognition based on transfer learning. In *Journal of Physics: Conference Series*, volume 1846, page 012012. IOP Publishing, 2021.

[13] Hiroaki Inotsume and Takashi Kubota. Terrain traversability prediction for off-road vehicles based on multi-source transfer learning. *ROBOMECH Journal*, 9(1):6, 2022.

[14] Jiashuo Liu, Zheyan Shen, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*, 2021.

[15] Julian Katz-Samuels, Julia B Nakhleh, Robert Nowak, and Yixuan Li. Training ood detectors in their natural habitats. In *International Conference on Machine Learning*, pages 10848–10865. PMLR, 2022.

[16] Haoyue Bai, Gregory Canal, Xuefeng Du, Jeongyeol Kwon, Robert D Nowak, and Yixuan Li. Feed two birds with one scone: Exploiting wild data for both out-of-distribution generalization and detection. In *International Conference on Machine Learning*, pages 1454–1471. PMLR, 2023.

[17] Adrian Schwaiger, Poulami Sinhamahapatra, Jens Gansloser, and Karsten Roscher. Is uncertainty quantification in deep learning sufficient for out-of-distribution detection? *Aisafety@ ijcai*, 54, 2020.

[18] Zhilin Zhao, Longbing Cao, and Kun-Yu Lin. Supervision adaptation balancing in-distribution generalization and out-of-distribution detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[19] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

[20] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in neural information processing systems*, 33:21464–21475, 2020.

[21] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*, 2018.

[22] Aristotelis-Angelos Papadopoulos, Mohammad Reza Rajati, Nazim Shaikh, and Jiamian Wang. Outlier exposure with confidence control for out-of-distribution detection. *Neurocomputing*, 441:138–150, 2021.

[23] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. *arXiv preprint arXiv:1711.09325*, 2017.

[24] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021.

[25] Nanyang Ye, Kaican Li, Haoyue Bai, Runpeng Yu, Lanqing Hong, Fengwei Zhou, Zhenguo Li, and Jun Zhu. Ood-bench: Quantifying and understanding two dimensions of out-of-distribution generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7947–7958, 2022.

[26] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[27] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[29] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

[30] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

[31] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018.

[32] Murat Sensoy, Lance Kaplan, and Melih Kandemir. Evidential deep learning to quantify classification uncertainty. *Advances in neural information processing systems*, 31, 2018.

[33] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

[34] Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*, 2017.

[35] Andrey Malinin. *Uncertainty estimation in deep learning with application to spoken language assessment*. PhD thesis, University of Cambridge, 2019.

[36] Kari Sentz and Scott Ferson. Combination of evidence in dempster-shafer theory. 2002.

[37] Audun Jøsang. *Subjective logic*, volume 3. Springer, 2016.

[38] Wentao Bao, Qi Yu, and Yu Kong. Evidential deep learning for open set action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13349–13358, 2021.