Terrain Costmap Generation via Scaled Preference Conditioning

Luisa Mao[1], Garrett Warnell[1,2], Peter Stone[1,3], Joydeep Biswas[1,4]

***Abstract*— Successful autonomous robot navigation in off-road domains requires the ability to generate high-quality terrain costmaps that are able to both generalize well over a wide variety of terrains and rapidly adapt relative costs at test time to meet mission-specific needs. Existing approaches for costmap generation allow for either rapid test-time adaptation of relative costs (e.g., semantic segmentation methods) or generalization to new terrain types (e.g., representation learning methods), but not both. In this work, we present *scaled preference conditioned all-terrain costmap generation* (SPACER), a novel approach for generating terrain costmaps that leverages synthetic data during training in order to generalize well to new terrains, and allows for rapid test-time adaptation of relative costs by conditioning on a user-specified *scaled preference context*. Using large-scale aerial maps, we provide empirical evidence that SPACER outperforms other approaches at generating costmaps for terrain navigation, with the lowest measured *regret* across varied preferences in five of seven environments for global path planning.**

I. Introduction

Off-road navigation requires terrain understanding that is both flexible and robust. Robots must (1) recognize diverse, even novel or mixed terrains, and (2) adapt traversal costs at test time for mission goals. Existing costmap methods struggle to achieve both generalization and adaptability simultaneously.

Approaches based on semantic segmentation support test-time variation of relative costs by assigning labels to terrain types and allowing users to directly adjust cost values whenever they wish. However, these methods are typically restricted to a fixed, predefined set of terrain classes, limiting their effectiveness in unfamiliar or ambiguous environments [1]–[3]. Conversely, representation learning approaches operate over feature spaces that allow them to better generalize to out-of-distribution inputs, but they typically lack mechanisms for easily adjusting relative costs at test time, making it difficult for users to express preferences or adapt to shifting objectives mid-mission [4]–[6].

In this work, we introduce *scaled preference conditioned all-terrain costmap generation* (SPACER), a novel method for terrain costmap generation that uniquely supports user-specified preferences over terrain pairs (Fig. 1). SPACER is conditioned on a novel, flexible, and expressive *scaled preference context*—a variable-length list of terrain image pairs annotated with a numerical preference strength—enabling nuanced, test-time control over terrain costs. Unlike prevalent preference-based inverse reinforcement learning (PbIRL) methods, which learn a reward function offline, our approach

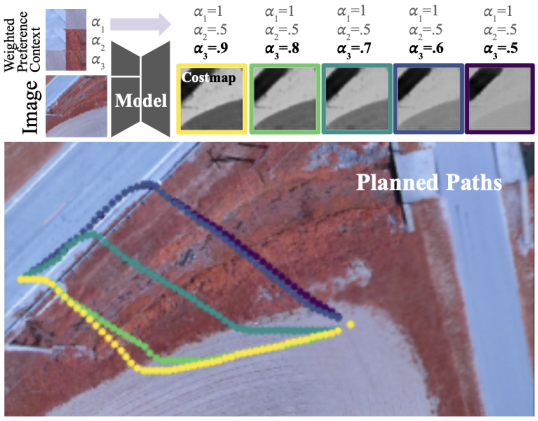

Fig. 1: Given an input image and preference over terrains, the model outputs a costmap. A scaled preference context is a set of pair comparisons of terrains (left patch preferred over right), scaled by a *strength of preference* α. Varying the strength of a preference leads to different gradation of costs and resulting planned paths.

learns *how* to perform PbIRL at runtime. During inference, the model receives only a small number of preferences, using them as contextual cues to infer mission-specific reward structures. This capability emerges from training on a wide variety of preference types and their corresponding reward functions, enabling the model to relate new preferences to appropriate reward functions at test time. Our method allows human operators to specify any number of scaled preferences over pairs of terrain patches by simply clicking on regions of an image and specifying the strength of preference for that pair. In addition to this input modality, SPACER generalizes well to novel and complex terrains by training on a large, diverse set of synthetic terrains and leveraging internet-pretrained visual representations. For example, if a last-mile delivery robot struggles with tall grass, this method enables rapid preference adjustments toward low grass or pavement, eliminating the need for retraining.

We evaluate SPACER across a range of aerial environments and preferences, and demonstrate that it consistently outperforms existing methods at generating mission-aligned costmaps. To further validate generalization and real-world applicability, we conduct both quantitative and qualitative evaluations on the RELLIS-3D dataset [1]—a high-resolution, multimodal benchmark for off-road, unstructured environments with diverse terrain types and fine-grained semantic labels. Additionally, we present ablation studies that isolate the impact of key architectural and training design choices.

II. Previous Work

Our work sits at the intersection of terrain-aware navigation and preference-based inverse reinforcement learning (PbIRL). Below, we review previous work in each area.

[1] The University of Texas at Austin, Austin, TX, USA.
[2] DEVCOM Army Research Laboratory, Austin, TX, USA.
[3] Sony AI, Boston, MA 02129, USA.
[4] NVIDIA, Santa Clara, CA 95051, USA.

A. Terrain-Aware Navigation via Costmaps

Our method transforms terrain images into costmaps usable by planners for terrain-aware navigation. Existing approaches generally fall into two categories: those using semantic representations and those using learned embeddings of terrain appearance.

Semantic approaches, such as classical or open-set semantic segmentation [2], [3], [7], assign terrain labels and then map these labels to user-defined costs. While this enables rapid test-time cost adjustment, classical methods are limited to predefined terrain classes, and open-set variants remain biased toward visually canonical terrains (e.g., green grass, gray concrete), which may fail in unfamiliar environments.

Embedding-based methods instead represent terrain as points in a learned feature space, removing the need for fixed semantic labels. However, because this space is not directly interpretable, cost functions must be learned through demonstrations or preferences. Most such methods require retraining whenever user preferences change [4]–[6], [8], [9] or restrict how relative costs can be tuned [10].

B. Preference-based Inverse Reinforcement Learning

Our approach builds on the framework of preference-based inverse reinforcement learning (PbIRL) [11], which learns reward (or cost) functions from user preferences. Traditional PbIRL methods learn these reward functions offline from large, fixed datasets and are applied in domains such as video games [12] and robotics [13]. In contrast, SPACER performs PbIRL at runtime: during inference, the model receives only a small number of scaled preferences and uses them as contextual cues to infer a task-specific reward function. This ability arises from training on a wide variety of preference types and their corresponding reward functions, enabling generalization from few preferences at test time.

Closest to our work, [14] and [10] also infer terrain costs from user preferences over terrain pairs, but they do not model preference scale. While prior work such as [15] encodes varying feedback noise through the rationality coefficient of a Boltzmann model, our scalar instead expresses preference certainty, producing stronger cost separation when the user expresses higher confidence.

C. Summary and Motivation

Existing approaches do not support both **deployment in unseen environments** and **test-time modulation of terrain costs according to user preferences**—particularly those expressed with scalable strength. SPACER addresses both limitations.

III. Problem: Preference-Aligned Path Planning

In this section, we formalize the preference-aligned path planning problem as introduced by prior work, which serves as the problem setting for our method described in Sec. IV. We focus on the SE(2) path planning problem, where the goal is to find a trajectory of poses $\Gamma = \{x_1, \ldots, x_n\}$, $x_i \in \mathcal{X}$, that minimizes a cost function C:

$$\Gamma^* = \arg\min_{\Gamma} \sum_i C(\Gamma_i) \quad \text{s.t.} \quad \begin{aligned} &x_1 = x_{\text{start}}, \\ &x_n = x_{\text{goal}}, \\ &\|x_{i+1} - x_i\| \le \delta, \ \forall i. \end{aligned}$$

where δ is a maximum step length determined by the discretization or dynamics.

In the terrain-aware preference-aligned setting, C is shaped by a black-box human preference function H over terrains $\tau \in \mathcal{T}$, mapping each terrain to a non-negative scalar cost $H(\tau) \in \mathbb{R}^{0+}$. These preferences capture subjective factors—e.g., some terrains may be physically damaging to the robot or undesirable for social or aesthetic reasons. Additionally, the robot does not directly observe terrains but receives visual input via an observation function $O(\mathcal{X}, T) = I$ that produces an image from a given pose and the global terrain map T. Prior work, PACER [10], introduced a costmap generator R that implicitly models H such that costmap $C = R(O(x, T)|H)$, and provided a method for approximating R and H, where H **and** $\mathcal{T}$ **could be unseen during training**.

PACER showed that a *necessary* condition to model H was to preserve the ordering of terrains according to preferences. Let x_τ denote a pose on terrain τ and $C|_x \in \mathbb{R}^{0+}$ the costmap evaluated at pose x. Given preference $\tau_a \succ \tau_b$ from H denoting terrain τ_a is preferred to τ_a, the cost at x_{τ_b} is less than that at x_{τ_b}:

$$\text{if } \tau_a \succ \tau_b, \quad \tau_a, \tau_b \in \mathcal{T} \quad \text{then} \quad C\big|_{x_{\tau_a}} < C\big|_{x_{\tau_b}} \tag{1}$$

However, this condition doesn't capture the magnitude of preferences, which are also important for planning. In this work, we aim not only to recover preferences but also to preserve their strength in the resulting cost estimates. Let $\tau_a \succ^{\alpha_{ab}} \tau_b$ denote a preference weighted by $\alpha_{ab} \in [0, 1]$. A small α_{ab} (i.e. a weak preference) should result in a smaller cost difference in R conditioned on H and stronger preferences should yield larger differences:

$$\begin{aligned} &\text{if } \tau_a \succ^{\alpha_{ab}} \tau_b, \ \tau_c \succ^{\alpha_{cd}} \tau_d, \ \alpha_{ab} < \alpha_{cd}; \ \tau_{a,b,c,d} \in \mathcal{T}, \\ &\text{then } \|C|x_{\tau_a} - C|x_{\tau_b}\| < \|C|x_{\tau_c} - C|x_{\tau_d}\|. \end{aligned} \tag{2}$$

Thus, if the human shows a much stronger preference between the terrains at poses x_{τ_c} and x_{τ_d} than between those at x_{τ_a} and x_{τ_b}, then the costmap $C = R(O(\cdot, T) \mid H)$ should reflect this difference by assigning a correspondingly larger cost difference between $x_{\tau_c}$and x_{τ_d} than x_{τ_a} and x_{τ_b}.

This section reviewed the preference-aligned path planning framework that motivates our work. Section IV presents our proposed costmap generation method, followed by dataset curation and training details in Section V.

IV. Method: Preference-Aligned All-Terrain Costmap Generation

We now describe *scaled preference conditioned all-terrain costmap generation* (SPACER), our approach to terrain costmap generation that both **generalizes well to new types of terrain** and also allows for **rapid test-time adaptation**

to new relative costs, which builds upon the problem formulation introduced in Section III. First, we define and connect to the notion of terrain cost a *scaled preference context*, a novel form of human input that allows users to specify not only their preferences between two options but also a scalar-valued weight indicating the *strength* of that preference. Next, we motivate and describe the neural network architecture that we adopted for SPACER.

A. Scaled Preference Context

We ask users to provide a *scaled preference*, which we define to be both a preferred option between to alternatives (here, terrains) and also a scalar-valued *strength* of that preference, where a larger strength communicates a stronger preference for the preferred option. More formally, we define a scaled preference as a tuple $\xi = (\tau^{(1)}, \tau^{(2)}, \alpha)$, which denotes that terrain $\tau^{(1)}$ is preferred to $\tau^{(2)}$ (lower costs as more preferred) with strength $\alpha \in [0, 1]$.

To model the underlying terrain costs from a scaled preference ξ, we employ the Bradley–Terry model [16], a well-established framework for reasoning over pairwise comparisons. The forward Bradley–Terry model relates pairwise match probabilities to costs, while the inverse model infers latent costs from given probabilities. It is suitable in our case because it allows us to infer the hidden human cost function H consistent with given preference strengths α.

Given the user-provided strength α of how much $\tau^{(1)}$ is preferred to $\tau^{(2)}$, the *inverse* Bradley-Terry model can be used to infer the costs $H(\tau^{(2)})$ and $H(\tau^{(1)})$ assigned by the hidden human preference function. Since $\tau^{(1)}$ is preferred to $\tau^{(2)}$ as given in the ordered tuple ξ, the probability $P(H(\tau^{(2)}) > H(\tau^{(1)}))$ is in [0.5, 1] so we map the strength $\alpha \in [0, 1]$ to $[0.5, 1]$ when converting it to a probability. **Succinctly, we interpret $\frac{\alpha+1}{2}$ to be the probability** $P(H(\tau^{(2)}) > H(\tau^{(1)}))$. The strength α and H are related by the *forward* Bradley Terry model, as shown below:

$$\frac{\alpha + 1}{2} = \frac{\exp\{H(\tau^{(2)})\}}{\exp\{H(\tau^{(1)})\} + \exp\{H(\tau^{(2)})\}} \tag{3}$$

$$= \sigma(H(\tau^{(2)}) - H(\tau^{(1)})) \tag{4}$$

Thus, given a sample $\xi = (\tau^{(1)}, \tau^{(2)}, \alpha)$, the *inverse* Bradley–Terry model can be applied to recover an estimate hidden preference function H. **Relating to condition 2, larger user-supplied αs lead to larger differences in costs.**

In this framework, the strength parameter α acts as a control variable, allowing the user to modulate the relative cost difference between terrains in the inferred model. This formulation ensures that stronger preferences correspond to larger cost differences in the recovered latent function H, and provides a method for recovering terrain costs from human preference data. We refer to an unordered collection of K scaled preferences as a *scaled preference context* $\Xi = (\xi_1, \ldots, \xi_K)$.

Therefore, our approach to predicting costmap $\hat{C}$ is a conditional inference problem given by

$$R(I|H) \approx \hat{R}_\phi(I|\Xi) = \hat{C} \tag{5}$$

where $\hat{R}_\phi$ is a neural network parameterized by ϕ, which implicitly learns the inverse Bradley Terry model given Ξ.

B. Model Architecture

We model the terrain cost function as a neural network that maps a terrain image to a costmap conditioned on the scaled preference context (Fig. 2). Our model consists of a *scaled preference context encoder* F_Ξ, image encoder F_I, latent UNet U, and decoder D. We give a novel interpretation of preference strength as an interpolation in a latent space, with α controlling the tradeoff between v_{strong} and v_{weak} embeddings. v_{strong} indicates the preference in the ordered pair is strong, whereas v_{weak} indicates no preference. The model formulation is presented in equation 7. Equation 6 shows the construction of the *scaled preference context* embeddings.

Preference Context Encoder: The preference context encoder F_Ξ maps each terrain image patch in Ξ into the latent space of the image encoder F_I. For each tuple $(\tau_i^{(1)}, \tau_i^{(2)}, \alpha_i)$, both terrain patches are encoded with F_I, and the preference strength α_i is embedded as a weighted combination of v_{strong} and v_{weak}:

$$F_\Xi(\Xi) = \left[F_I(\tau_i^{(1)}),\ F_I(\tau_i^{(2)}),\ \alpha_i v_{\text{strong}} + (1-\alpha_i) v_{\text{weak}}\right]_{i=1}^{K} \tag{6}$$

The injection of the preference strength α into the network is a core innovation of our architecture. Instead of passing α directly as a scalar, we map it into a latent embedding space via a convex combination of two learned embeddings, v_{strong} and v_{weak}, denoting strong and neutral preferences: $v_{\text{strong}} \cdot (\alpha) + v_{\text{weak}} \cdot (1 - \alpha)$. Our design is inspired by the time embeddings used in diffusion models [17], which have proven effective for conditioning on continuous variables. The patch latents are concatenated channel-wise with this *strength embedding*, providing the model with a rich conditioning signal.

Pretrained Encoder and Decoder: The costmap $\hat{C}$ is predicted from the input image I and the preference context Ξ. We use F_I and D as the pretrained VAE encoder and decoder from Stable Diffusion [17], with weights frozen during training, and U as a latent conditional UNet:

$$\begin{aligned} \hat{C} &= \hat{R}_\phi(I \mid \Xi) \\ &= D\Big(U\left[F_I(I),\ F_\Xi(\Xi)\right]\Big) \end{aligned} \tag{7}$$

Latent UNet: The latent UNet incorporates the conditioning embedding with the image embedding through cross attention layers, so the preference context may have a variable length. The outputs of the UNet are latent vectors which are decoded by the frozen VAE decoder D into a local BEV costmap. The mean of the three output channels is interpreted as the costmap $\hat{C}$.

V. Dataset Curation and Training

Since collecting large-scale human preference data is challenging, we procedurally synthesize a dataset consistent with

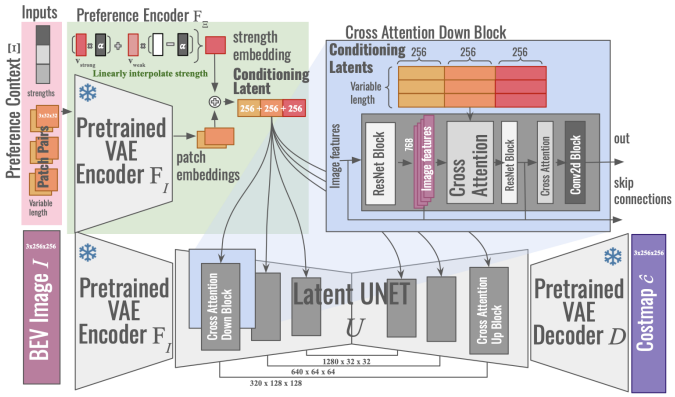

Fig. 2: Model architecture. Inputs are BEV image and variable-length preference context consisting of patch pairs and strengths. The conditioning latents are the concatentation of patch embeddings and strength embedding (calculated as a linear interpolation of *strong* and *weak* embeddings). The output is a costmap. updated

our preference model to enable scalable training (further described in Sec. V-B). Our dataset $\mathcal{D} = \{(I_i, \Xi_i, S_i, C_{T_i})\}_{i=1}^n$ consists of an image I, scaled preference context Ξ, list of segmentation masks S, and target costmap C_T. Terrain classes are only used during training. We do not require a global set of terrain classes during deployment.

A. Loss Function

Our objective is to learn a costmap prediction $\hat{C}$ that (1) is consistent with human preferences over terrain and (2) well-regularized.

First, by $\mathcal{L}_1$, the model outputs are encouraged to match the given preference context Ξ. To extract the costs of k terrain classes, we use segmentation masks $S = \{s_1, \ldots, s_k\}$, where $\mu_s(C)$ denotes the mean over mask s of costmap C. By the forward Bradley-Terry model, we match the log-odds calculated from the predicted costmap $\mu_{s_i^{(1)}}(\hat{C}) - \mu_{s_i^{(2)}}(\hat{C})$ to the target log-odds given in the preference context $\sigma^{-1}(\frac{\alpha_i+1}{2})$ to ensure predicted costs are consistent with the user-provided preference.

Second, by $\mathcal{L}_2$, we introduce an additional term that anchors the predicted costs to a target distribution, ensuring they remain within the range $(0, 1)$ (see Section VI-C for ablation results). $\mathcal{L}_2$ prevents drift, since relative preferences are invariant under a uniform scaling of costs, optimizing only for preference consistency can cause the learned costmap to drift in magnitude.

For a given scaled preference context Ξ, predicted costmap $\hat{C}$, and target costmap C_T, we define the loss as:

$$\mathcal{L} = \mathcal{L}_1 + \lambda \mathcal{L}_2, \tag{8}$$

$$\mathcal{L}_1 = \sum_{i \le |\Xi|} \mathcal{L}_\delta \left(\mu_{s_i^{(2)}}(\hat{C}) - \mu_{s_i^{(1)}}(\hat{C}),\ \sigma^{-1}\left(\frac{\alpha_i + 1}{2}\right)\right), \tag{9}$$

$$\mathcal{L}_2 = \mathcal{L}_\delta(\hat{C}, C_T), \tag{10}$$

where λ tunes the relative strength between $\mathcal{L}_1$ and $\mathcal{L}_2$, $\mathcal{L}_\delta$ is the Huber loss, and σ^{-1} is the inverse sigmoid. We additionally train with a low learning rate of 1e-5 and gradient accumulation. Fig. 3 shows the loss function.

B. Data Curation

Each training example is curated to be *consistent with the Bradley-Terry model* of pairwise preferences.

Specifically, for each ordered preference triplet $(\tau^{(1)}, \tau^{(2)}, \alpha) \in \Xi$, the preference strength $\alpha \in [0, 1]$ is computed based on the softmax-derived probability that $\tau^{(2)}$ has a higher cost than $\tau^{(1)}$ under the ground-truth costmap C_T. Since $\tau^{(1)}$ is more preferred (lower cost) than $\tau^{(2)}$, the range of this probability is [0.5, 1] which we remap to [0,1] for α.

We define:

$$\alpha = 2\left(\frac{\exp[\mu_{s^{(2)}}(C_T)]}{\exp[\mu_{s^{(1)}}(C_T)] + \exp[\mu_{s^{(2)}}(C_T)]}\right) - 1$$

.

To train and evaluate our model, we procedurally generate a synthetic dataset consisting of terrain images, ground-truth costmaps, and preference contexts. The dataset is built using a bank of pre-defined terrains and scalar cost values, combined using spatial masks. We provide an ablation study on real versus synthetic data in Section VI-C. The full generation process is outlined below (with steps 5,6 shown in Fig 3).

1) Define a bank of terrain types $\mathcal{T} = \{\tau_1, \tau_2, \ldots, \tau_n\}$ and global scalar costs $G = \{\gamma_1, \ldots, \gamma_m\}$.
2) Generate k segmentation masks $S = \{s_1, \ldots, s_k\}$ using procedural generation (e.g., the diamond-square algorithm [18]).
3) Sample k terrains $\{\tau_1, \ldots, \tau_k\} \subset \mathcal{T}$ and k cost values $\{\gamma_1, \ldots, \gamma_k\} \subset G$.
4) Assign each terrain and cost to a mask, forming triplets $\{(s_1, \tau_1, \gamma_1), \ldots, (s_k, \tau_k, \gamma_k)\}$.
5) Use these assignments to construct the input image I and the ground-truth costmap C_T.
6) From the assigned triplets, construct a scaled preference context:

$$\Xi = \left\{\left(\tau_i^{(1)}, \tau_i^{(2)}, 2\sigma(\gamma_i^{(2)} - \gamma_i^{(1)}) - 1\right)\right\}_{i=1}^K$$

7) Return I, Ξ, S, and C_T.

The dataset contains 12 images of terrains ranging from `sand`, `mud`, `forest floor`, `river pebbles`, `grass`, `snow`, and `concrete` among others. Crucially, the use of a pretrained variational autoencoder (VAE) trained on internet-scale data allows the model to already encode strong priors about the structure and statistics of realistic natural images. As a result, our model can learn effectively from a relatively small number of synthetic BEVs, since the VAE provides a perceptual space aligned with the distribution of real-world visual inputs. This is helpful since size of the dataset is combinatorial with the number of BEV images.

VI. Experiments

To evaluate SPACER, we seek to answer the following questions: **(Q1)** Does SPACER generate accurate costmaps up to scale consistent with operator input? **(Q2)** Do paths planned over costmaps generated by SPACER outperform

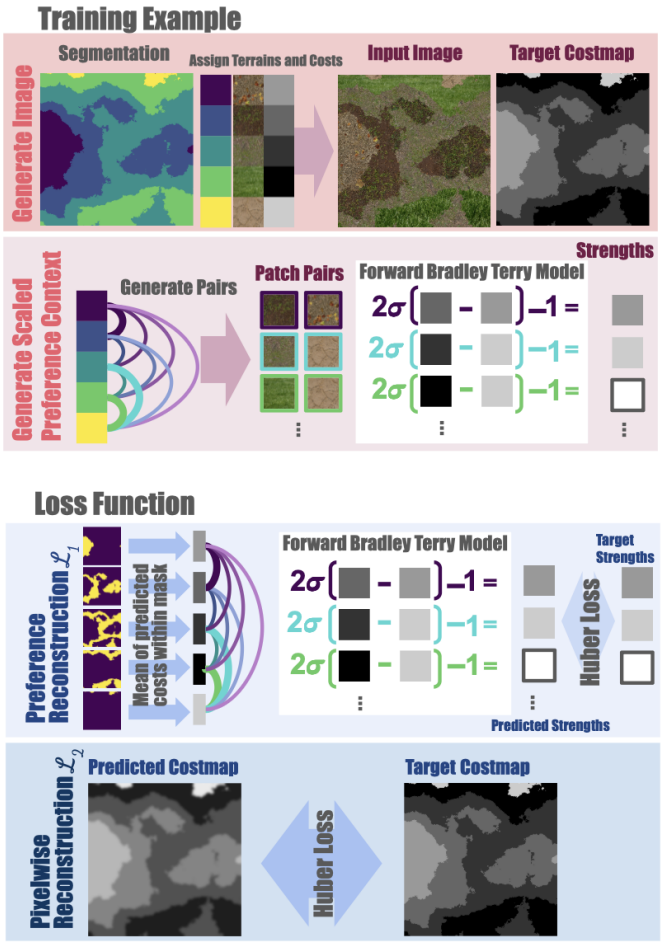

Fig. 3: The construction of the Target Costmap and Scaled Preference Context in the Training Example parallels the loss function. The loss function has two parts: *preference reconstruction* loss $\mathcal{L}_1$ and pixelwise reconstruction $\mathcal{L}_2$ against the target costmap. In $\mathcal{L}_1$, the preference context is constructed from the *predicted* costmap using the ground truth segmentation masks, and the *predicted* strengths are matched to the input (target) strengths. The $\mathcal{L}_2$ term keeps costs from exploding while the $\mathcal{L}_1$ term is critical for the model's preference-alignment ability. Note: $\mathcal{L}_1$ is shown in an equivalent form here for clarity, but we implement it as in eq. 9.

competing methods in terms of Hausdorff distance to ground-truth paths and the introduced metric of path *regret*, which measures additional traversal cost compared to optimal paths? **(Q3)** How do our design choices of loss function and synthetic training data affect performance?

Unfortunately, no other method operates on the same space of inputs, so we provide classifier methods with more information–absolute costs for all classes–which users may not have in actual deployments.

We use different datasets and baselines for **(Q1)** and **(Q2)** due to their distinct goals. **(Q1)** evaluates whether SPACER generates accurate costmaps up to scale, which requires ground-truth terrain labels. Since large-scale aerial datasets lack such pixel-level annotations—especially in off-road settings—we use the RELLIS-3D dataset [1], which includes labeled terrain classes, and compare against segmentation-based baselines and the zero-shot method PACER. In contrast, **(Q2)** focuses on the planning utility of the generated costmaps in real-world, large-scale environments. Large-scale planning demands global aerial maps, which are typically unlabeled and unsuitable for supervised training. Therefore, we evaluate planning performance using these maps and compare against pretrained, open-set foundation models capable of generalizing without task-specific fine-tuning.

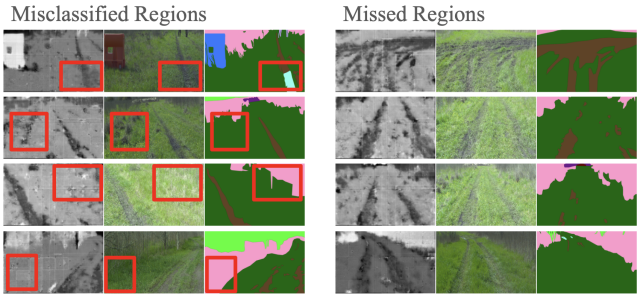

Fig. 4: Examples of misclassified segmentation masks (left) and missed regions in segmentation masks (right). Each triplet consists of a SPACER costmap, RELLIS-3D image, and RELLIS-3D segmentation mask. We find that the RELLIS-3D segmentation masks are not accurate and fine enough for a quantitative evaluation across the whole dataset.

Since the costs from SPACER are correct up to scale, we normalize the local costmap C by matching the lowest and highest predicted cost to the lowest and highest ground truth cost for all experiments.

A. Accuracy of Generated Costmaps

Exploring **(Q1)**, we evaluate SPACER on the RELLIS-3D dataset [1] with three preferences P, and report the MAE to show how closely the varying preferences are adhered to.

Methods and Baselines: We compare against against PACER and dense segmentation models. First, since PACER does not support direct encoding of relative preference strengths, we approximate its behavior by providing only ordered terrain label pairs. Additionally, PACER requires a training phase on real-world data to learn a reasonable prior, due to its limited context length for expressing preferences. In contrast, SPACER is trained entirely on synthetic data (see Section V-B) and does not require task-specific fine-tuning at deployment time. The second baseline is a supervised segmentation model built on a pretrained DEEPLABV3_RESNET50 backbone. We train two variants of this model: one fine-tuned on the RELLIS-3D dataset and the other on the same synthetic dataset used to train SPACER. The model trained on RELLIS-3D represents an upper-bound baseline, as it is trained and evaluated on the same domain. The synthetic-trained version provides a direct comparison point for SPACER under equivalent data conditions. While both methods use the same training data, SPACER significantly outperforms the segmentation model trained on synthetic data. This performance gap highlights the advantage of preference-driven costmap inference over traditional semantic segmentation, particularly in scenarios where semantic classes may not align with terrains traversed.

Experimental Setup: As we find there are misclassifications in many images (shown in Fig. 4 and futher discussed in Section VII), we provide an evaluation over a subset of 860 images (cropped to remove the sky, and resized). **Metrics:** We report the mean absolute error (pixel-wise difference) per costmap for each method evaluated on three terrain class

orderings, as well as the class-wise MAE for the three most frequent classes in Tab. I.

Results and Discussion: The Segmentation (Ground Truth) baseline performs best overall (Table I), as it was trained directly on a subset of the RELLIS-3D dataset. The remaining three methods exhibit higher MAE due to discrepancies between the dataset's provided ground-truth segmentation masks and the actual terrain textures in the images. Among these, SPACER consistently achieves the lowest total MAE and frequently the lowest class-wise MAE across all three evaluated preference orderings. PACER shows slightly higher MAE—while it is also preference-aware, its limited preference context length cannot fully capture all relevant comparisons for the many terrain classes present in RELLIS-3D. As a result, it leans more on learned priors over terrain and preference distributions, occasionally causing severe errors on underrepresented classes not shown above. Segmentation performs worst, constrained by the ontology of the synthetic dataset it was trained on. Unlike SPACER, it fails to generalize terrain understanding or preference modeling to the RELLIS-3D domain. These findings largely match our expectations: representation-learning methods (i.e. PACER) cannot modulate costs as flexibly and classification-based (i.e. segmentation baseline) methods cannot handle visually out-of-distribution terrains. **Answering (Q1)**, we find that SPACER generates accurate costmaps that preserve relative cost magnitudes based on the given preference ordering and strengths.

B. Accuracy of Planned Paths

In this experiment, we explore **(Q2)**: whether SPACER produces planned paths that better adhere to operator preferences than methods that ignore preference weights or require known classes.

Methods and Baselines: We evaluate SPACER against the following baselines: PACER, ClipSeg (weighted), and ClipSeg (argmax). ClipSeg (weighted) takes the linear combination of cost assignments to labels, weighted by the activation of the class label while ClipSeg (argmax) assigns cost based only on the most probable label. Both of these methods of open-set cost assignment are common, and each have limitations. ClipSeg (weighted) tends to produce costmaps that are smoother but less spatially precise in complex environments, while ClipSeg (argmax) yields sharper boundaries but can lead to high-variance costmaps. Qualitative examples are shown in 5.

Experimental Setup: We implemented a simulator which performs A* planning on aerial drone maps with a 40-connected angular lattice, where each node considers 40 discretized heading directions to enable fine-grained trajectory generation. This planner is fixed across all baselines to ensure comparability. In each environment, we deploy the robot with a fixed terrain ordering in their preference, but vary the distances between the costs of terrains (i.e. vary the strengths). **Metrics:** We evaluate each method using the Hausdorff distance between planned paths on predicted vs. ground truth costmaps, and two forms of *regret*. The

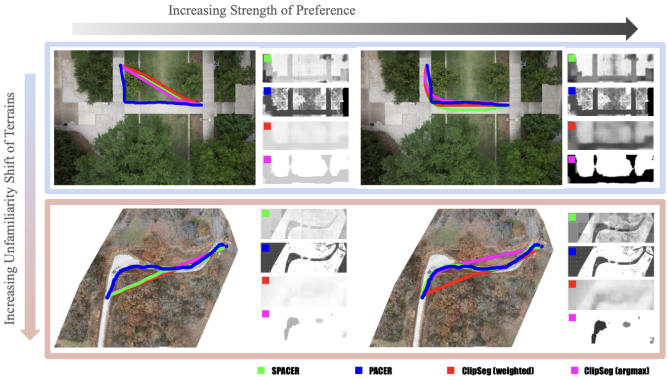

Fig. 5: Qualitative comparison of costmaps generated by all methods. As preference strength varies across rows, PACER cannot adapt. As terrain *visual* appearances become more misaligned with their classes across columns (i.e. vegetation is not green), classification baselines cannot adapt. SPACER produces costmaps and planned paths which are most aligned with the userspreferences

Hausdorff distance captures spatial deviation between paths, while regret quantifies suboptimality in cost. Let $\bar{\Gamma}$ and Γ denote the paths planned on ground truth and predicted costmaps, respectively, with cost functions $\bar{C}(\cdot)$, $C(\cdot)$. Then $\rho^* = \bar{C}(\Gamma) - \bar{C}(\bar{\Gamma})$ and $\hat{\rho} = C(\bar{\Gamma}) - C(\Gamma)$ represent *regret* on the ground truth and predicted costmaps respectively. ρ^* measures how optimal the predicted path is relative to the ground truth path, while $\hat{\rho}$ reflects how well the model evaluates the ground truth path under its own predicted costmap. We report the mean Hausdorff distance, ρ^*, and $\hat{\rho}$ across deployments for each environment.

Results and Discussion: We find that SPACER achieved the lowest Hausdorff distance in two environments and the second-lowest in another three environments, for the overall best performance, as shown in Table II. From Table III, we find that SPACER achieved the lowest ρ^* in five out of the seven environments. PACER performs better on ρ^* in E5 and E6, which we attribute to environment-specific variation or sensitivity to initialization rather than a systematic shortcoming of SPACER. SPACER also achieved the lowest $\hat{\rho}$ in three environments.

PACER generally exhibited high $\hat{\rho}$ because it cannot adjust relative terrain costs, as shown in IV. As a result, it often considers the ground-truth path to be suboptimal when it traverses mildly unpreferred terrains, even though doing so is more efficient in terms of path length.

Open-set segmentation models showed high ρ^*, particularly in out-of-distribution (OOD) environments E4–E7. These models tend to struggle in novel settings due to mismatches between their learned visual concepts and the actual appearance of terrain in unfamiliar domains. This often led to low-confidence predictions or misclassification, causing the planner to choose suboptimal routes that traverse undesirable terrain and incur high regret over longer trajectories.

Interestingly, ClipSeg (weighted) had consistently low $\hat{\rho}$ across all environments, likely because its output maps were low-confidence and blurry—effectively assigning similar costs throughout. As a result, the model evaluated the ground-truth path as only marginally worse than its own, despite producing poor predicted paths.

TABLE I: Mean Absolute Error (lower is better) for each method under different preferences across selected semantic classes. Best is bolded. Second best is italicized. Asterisk * denotes the method is given *more* information.

Preference	Method	Total MAE	Grass MAE	Bush MAE	Mud MAE	Barrier MAE
P1	Segmentation (Ground Truth) *	0.0372	0.0197	0.0429	0.1724	0.1270
	SPACER	**0.1055**	**0.0711**	**0.1724**	**0.2215**	**0.0021**
	PACER	*0.1374*	*0.1041*	*0.2059*	*0.2734*	*0.3105*
	Segmentation *	0.3003	0.2970	0.2732	0.3178	0.7677
P2	Segmentation (Ground Truth) *	0.0359	0.0205	0.0424	0.1699	0.0323
	SPACER	**0.1035**	**0.0860**	**0.1193**	*0.2218*	**0.0000**
	PACER	*0.1611*	*0.1571*	*0.1545*	**0.2142**	0.3451
	Segmentation *	0.2914	0.3004	0.2700	0.3168	*0.3210*
P3	Segmentation (Ground Truth) *	0.0256	0.0140	0.0289	0.1296	0.0996
	SPACER	**0.1412**	**0.0501**	**0.0360**	**0.0466**	**0.0007**
	PACER	*0.1661*	*0.1853*	*0.0791*	*0.0653*	*0.0768*
	Segmentation *	0.2951	0.2716	0.3607	0.3162	0.3050

Class distribution: Grass = 68%, Bush = 18%, Mud = 5%, Barrier = 2%

TABLE II: Hausdorff Distance (lower is better) between planned paths on ground truth and predicted costmaps. Best value is bolded and second-best is italicized. Asterisk * denotes the method is given *more* information. E1–E3 contain visually canonical terrains; E4–E7 contain non-visually canonical terrain.

Method	E1	E2	E3	E4	E5	E6	E7
SPACER	*14.78*	**5.48**	**7.89**	*14.96*	44.99	*68.02*	**40.37**
PACER	23.21	22.84	19.37	20.26	*37.24*	**30.22**	161.19
ClipSeg (weighted) *	**10.73**	13.53	*10.02*	51.21	**30.05**	100.77	193.55
ClipSeg (argmax) *	19.78	*13.29*	46.60	**14.83**	51.00	386.34	*64.57*

TABLE III: Performance evaluated on ρ^* (lower is better). E1–E3 contain visually canonical terrains; E4–E7 contain non-visually canonical terrain.

Method	E1	E2	E3	E4	E5	E6	E7
SPACER	**1.40**	**1.05**	**0.90**	**3.77**	25.50	*16.19*	**7.46**
PACER	*9.55*	*5.81*	2.82	*4.46*	**6.80**	**7.71**	71.67
ClipSeg (weighted) *	12.78	6.23	3.83	33.52	*13.29*	302.44	*60.15*
ClipSeg (argmax) *	9.56	8.13	*1.78*	40.66	71.58	241.27	63.43

Answering (Q2), we find that the preference-expression and generalizability of SPACER does lead to better planned paths than the baselines on a diverse set of aerial maps from different environments.

C. Ablations

To explore **(Q3)**, we train our model on fully real (from robot deployments around a college campus) or fully synthetic data, using variations of our loss functions. We report Mean Absolute Error against the ground truth costmap. Each model is trained on five deployments and evaluated on three held-out deployments in different environments.

Fig. 6 shows results from the ablation study. Rows represent realistic or synthetic training data, and columns represent the loss functions. In Fig. 6 and Tab. V, we include both the un-normalized and normalized (divided by the maximum) output from the $\mathcal{L}_1$-only model in our results, since the raw output values have large magnitudes. Training only on pixel-wise reconstruction losses such as Huberloss (as is done in some prior work [10]) results in poor performance, as it gives too weak a signal. The models trained only on $\mathcal{L}_1$ appear to follow the given preferences, though training is unstable and cost magnitudes tend to explode as the model is unregularized. Combining the losses, we weight each such that the magnitude of $\mathcal{L}_1$ is 5x greater than that of $\mathcal{L}_2$. The combined losses allow the model to learn human preferences while keeping output costs small and stable.

TABLE IV: Performance evaluated on $\hat{\rho}$ (lower is better). E1–E3 contain visually canonical terrains; E4–E7 contain non-visually canonical terrain.

Method	E1	E2	E3	E4	E5	E6	E7
SPACER	10.27	**2.03**	*2.70*	*15.39*	56.80	81.75	**17.16**
PACER	13.84	14.33	9.50	22.32	110.02	*61.18*	87.15
ClipSeg (weighted) *	**3.21**	*2.54*	**1.15**	15.58	*11.08*	**53.84**	*26.41*
ClipSeg (argmax) *	*3.78*	3.60	3.41	**14.00**	**9.72**	212.92	30.78

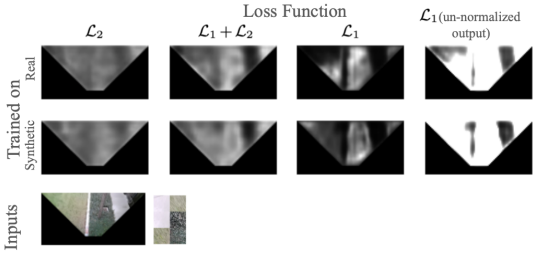

Fig. 6: Qualitative performances of various models, comparing models trained with Real vs. Synthetic data and different loss functions.

Training Data	$\mathcal{L}_2$	$\mathcal{L}_1 + \mathcal{L}_2$	$\mathcal{L}_1$	$\mathcal{L}_1$ **(un-normalized)**
Real Training Data	0.1516	**0.1157**	0.1350	1.0965
Synthetic Training Data	0.1459	**0.1205**	0.1391	1.2124

TABLE V: Mean Absolute Error (lower is better) averaged across three robot deployments in new environments, comparing performances of various models trained with Real vs. Synthetic data and different loss functions. $\mathcal{L}_1 + \mathcal{L}_2$ performs best, with little between *real* and *synthetic* data.

Further, we find that training on purely synthetic data is not detrimental compared to real training data, with synthetic data being much less expensive to obtain. Thus, we choose to train on synthetically-trained models with $\mathcal{L}_1 + \mathcal{L}_2$ loss.

D. Runtimes

Inference time for a single forward pass was benchmarked on an NVIDIA RTX A6000 GPU. PACER achieved the fastest runtime at 0.012 s per image, followed by DEEPLABV3_RESNET50 (0.029 s), SPACER (0.057 s), and CLIPSeg (0.058 s).

VII. FURTHER QUALITATIVE RESULTS

Using SPACER, we found that many images in the RELLIS-3D dataset either have coarse segmentations or mis-

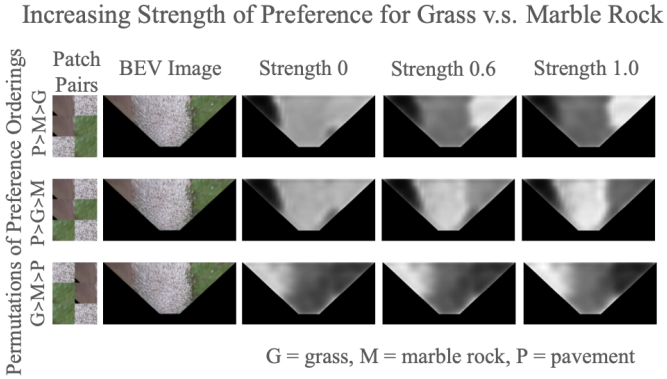

Fig. 7: Effects of increasing strength of preference of grass and marble rock (3rd pair in the context) versus preference orderings over terrains.

classified regions. We provide examples below of contention between the terrains and labeled masks. In addition to this uncertainty regarding where the segmentation boundaries should lie, we find that terrains within the same class label can vary widely in appearance (Fig. 4). These are limitations with segmentation approaches in general, so we provide a limited quantitative evaluation of this dataset as well as a qualitative evaluation, the latter of which we claim is more important for evaluating performance.

Qualitative results from varying preference orderings and strengths for a BEV image (collected during a robot deployment on a university campus) is shown in Fig. 7. Each row shows a different total ordering over three terrains (`pebble pavement`, `marble rock`, `grass`) as represented by the patch pairs. Each column of costmaps shows the model output at a different strength of the `grass` v.s. `marble rock` pair. The other two pairs have a strength of 1. At low strengths, grass and marble rock are given similar costs. At high strengths, there is strong contrast between the costs for grass and marble rock. In all cases, the cost ordering is consistent with the total ordering dictated by the patch pairs.

VIII. Conclusion

This paper introduced SPACER, a novel framework for zero-shot terrain-aware navigation that enables fine-grained user control over terrain cost magnitudes in novel, out-of-distribution environments. By addressing the limitations of fixed semantic ontologies and ingrained visual priors in existing segmentation and representation-learning approaches, SPACER introduces a more expressive and flexible formulation of user preferences. Our method leverages a new mathematical representation of preference intensity, a targeted synthetic data generation process, and a tailored loss function to train models that are both generalizable and responsive to user intent. In future work, we aim to extend SPACER beyond RGB-only input to include modalities such as depth or multimodal signals.

References

[1] P. Jiang, P. Osteen, M. Wigness, and S. Saripalli, "Rellis-3d dataset: Data, benchmarks and analysis," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 1110–1116.

[2] T. Guan, D. Kothandaraman, R. Chandra, A. J. Sathyamoorthy, K. Weerakoon, and D. Manocha, "Ga-nav: Efficient terrain segmentation for robot navigation in unstructured outdoor environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8138–8145, 2022.

[3] T. Lüddecke and A. Ecker, "Image segmentation using text and image prompts," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 7086–7096.

[4] X. Yao, J. Zhang, and J. Oh, "RCS: Ride Comfort-aware Visual Navigation via Self-supervised Learning," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 7847–7852.

[5] H. Karnan, E. Yang, D. Farkash, G. Warnell, J. Biswas, and P. Stone, "STERLING: Self-Supervised Terrain Representation Learning from Unconstrained Robot Experience," in *Proceedings of The 7th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 229. PMLR, 2023, pp. 2393–2413.

[6] K. S. Sikand, S. Rabiee, A. Uccello, X. Xiao, G. Warnell, and J. Biswas, "Visual representation learning for preference-aware path planning," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 11 303–11 309.

[7] X. Meng, N. Hatch, A. Lambert, A. Li, N. Wagener, M. Schmittle, J. Lee, W. Yuan, Z. Chen, S. Deng, G. Okopal, D. Fox, B. Boots, and A. Shaban, "TerrainNet: Visual Modeling of Complex Terrain for High-speed, Off-road Navigation," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023.

[8] J. Zürn, W. Burgard, and A. Valada, "Self-supervised Visual Terrain Classification from Unsupervised Acoustic Feature Learning," *IEEE Transactions on Robotics*, vol. 37, no. 2, pp. 466–481, 2020.

[9] A. Zhang, H. Sikchi, A. Zhang, and J. Biswas, "Creste: Scalable mapless navigation with internet scale priors and counterfactual guidance," *arXiv preprint arXiv:2503.03921*, 2025.

[10] L. Mao, G. Warnell, P. Stone, and J. Biswas, "pacer: Preference-conditioned all-terrain costmap generation," *IEEE Robotics and Automation Letters*, vol. 10, no. 5, pp. 4572–4579, 2025.

[11] C. Wirth, R. Akrour, G. Neumann, and J. Fürnkranz, "A survey of preference-based reinforcement learning methods," *Journal of Machine Learning Research*, vol. 18, no. 136, pp. 1–46, 2017. [Online]. Available: http://jmlr.org/papers/v18/16-634.html

[12] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/d5e2c0adad503c91f91df240d0cd4e49-Paper.pdf

[13] M. Schoenauer, R. Akrour, M. Sebag, and J.-C. Souplet, "Programming by feedback," in *Proceedings of the 31st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, E. P. Xing and T. Jebara, Eds., vol. 32, no. 2. Bejing, China: PMLR, 22–24 Jun 2014, pp. 1503–1511. [Online]. Available: https://proceedings.mlr.press/v32/schoenauer14.html

[14] M. Zucker, J. A. Bagnell, C. G. Atkeson, and J. Kuffner, "An optimization approach to rough terrain locomotion," in *2010 IEEE International Conference on Robotics and Automation*. IEEE, 2010, pp. 3589–3595.

[15] G. R. Ghosal, M. Zurek, D. S. Brown, and A. D. Dragan, "The effect of modeling human rationality level on learning rewards from multiple feedback types," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 5, pp. 5983–5992, Jun. 2023. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/25740

[16] H. Turner and D. Firth, "Bradley-terry models in R: The BradleyTerry2 package," *Journal of Statistical Software*, vol. 48, no. 9, pp. 1–21, 2012.

[17] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 10 684–10 695.

[18] A. Fournier, D. Fussell, and L. Carpenter, *Computer rendering of stochastic models*. New York, NY, USA: Association for Computing Machinery, 1998, p. 189–202. [Online]. Available: https://doi.org/10.1145/280811.280993