# LatentBKI: Open-Dictionary Continuous Mapping in Visual-Language Latent Spaces with Quantifiable Uncertainty

Joey Wilson\*, Ruihan Xu\*, Yile Sun, Parker Ewen, Minghan Zhu, Kira Barton, and Maani Ghaffari

Abstract—This paper introduces a novel probabilistic mapping algorithm, LatentBKI, which enables open-vocabulary mapping with quantifiable uncertainty. Traditionally, semantic mapping algorithms focus on a fixed set of semantic categories which limits their applicability for complex robotic tasks. Vision-Language (VL) models have recently emerged as a technique to jointly model language and visual features in a latent space, enabling semantic recognition beyond a predefined, fixed set of semantic classes. LatentBKI recurrently incorporates neural embeddings from VL models into a voxel map with quantifiable uncertainty, leveraging the spatial correlations of nearby observations through Bayesian Kernel Inference (BKI). LatentBKI is evaluated against similar explicit semantic mapping and VL mapping frameworks on the popular Matterport3D and Semantic KITTI data sets, demonstrating that LatentBKI maintains the probabilistic benefits of continuous mapping with the additional benefit of open-dictionary queries. Real-world experiments demonstrate applicability to challenging indoor environments.

*Index Terms*—Mapping, Semantic Scene Understanding, Deep Learning for Visual Perception.

## I. INTRODUCTION

**R**OBOTS require informative world models to autonomously navigate the world, commonly known as maps. Mapping methods represent the geometry of the robot's surroundings and often include semantic information relevant to robotic task success. While some works have proposed mapless autonomous navigation [1], [2], maps are commonly used in robotics due to the ability to leverage temporal information within an interpretable world model.

Maps are also capable of storing a high level of scene understanding with multi-modal information such as occupancy, semantics, traversability, and uncertainty. As deep neural networks have rapidly progressed, mapping algorithms have also evolved from binary occupancy grids [3] to model higher levels of scene understanding such as through semantic information [4]. However, real-world environments contain complex and detailed scenes that cannot be captured through closed-dictionary semantic maps.

Recently, deep learning has produced foundation models trained on large, varied data sets with the reported ability to generalize to out-of-distribution data, solving a limitation of

This paper was recommended for publication by Editor Aleksandra Faust upon evaluation of the Associate Editor and Reviewers' comments. This work was partly supported by the DARPA TIAMAT project.

\*The authors contributed equally.

(Corresponding Author: J. Wilson)

Digital Object Identifier (DOI): see top of this page.



Fig. 1: LatentBKI enables semantic mapping by leveraging open-dictionary language inference with vision-language (VL) model data. VL networks process exteroceptive data to generate point-wise features, which LatentBKI integrates into a 3D map via Bayesian Kernel Inference (BKI). Unlike prior VL mapping methods, LatentBKI updates nearby voxels using spatial information and maintains quantifiable uncertainty via conjugate priors. As shown, LatentBKI applies to real-world scenes (middle), quantifies semantic uncertainty per voxel (left), and decodes voxel features into categories using the VL network's language-driven decoder (right).

previous semantic segmentation neural networks [5], [6]. Of particular interest to the robotics community, Vision-Language (VL) models [7] extend vision processing to share a feature space with large language models, enabling applications such as open-dictionary semantic segmentation networks [8]. Following the success of open-dictionary segmentation networks, several robotic mapping papers have proposed to integrate open-dictionary segmentation features within a robotic map in order to enable language-based queries and navigation [9]-[13]. Within robotic open-dictionary mapping networks, a common approach is to fuse input points into the map through a simple moving average scheme, which does not require additional training [10], [11], [13]. Inspired by the success of the simple averaging technique, we propose to leverage the structure of continuous mapping to enable spatial smoothing and uncertainty quantification.

Continuous mapping is an approach to probabilistic mapping which leverages Bayesian inference and spatial context to create complete maps with uncertainty [14]–[16]. Quantifiable uncertainty is critical for robotics applications, as observations of the world are often limited and noisy, leading to errors and incomplete maps. Through uncertainty quantification in the form of variance, planning algorithms can identify potentially dangerous conditions [17] and the optimal trajectories to obtain new measurements [18], [19]. Additionally, in the real world data is often sparse whether from sensors or views, leading to incomplete map representations. An efficient approach to continuous mapping known as Bayesian kernel inference (BKI) leverages spatial context to update nearby unobserved

Manuscript received: October 3, 2024; Revised December 23, 2024; Accepted January 13, 2025.

The authors are with the University of Michigan, Ann Arbor, MI 48109, USA. {wilsoniv,rhxu,sunyyyl,pewen}@umich.edu {minghanz,bartonkl,maanigj}@umich.edu

voxels through the use of an extended likelihood function defined by a kernel [20]. BKI yields an efficient probabilistic update, however has not been studied as a possible solution for open-dictionary mapping.

In this paper we propose an extension of continuous mapping to the latent space of neural networks called LatentBKI, which enables open-dictionary continuous mapping with spatial smoothing and quantifiable uncertainty. Compared to prior works in open-dictionary mapping, our method also leverages a weighted average to calculate the per-voxel expectation, however enables uncertainty quantification and fills in gaps within the map through BKI [20]. Compared to prior research in continuous mapping for robotic maps, LatentBKI proposes a Gaussian likelihood conjugate pair for latent space measurements, enabling open-dictionary mapping and inference without any loss in performance.

We evaluate our method against VLMap, which leverages a moving average recursive approach and is directly comparable to the expectation produced by LatentBKI. We also quantitatively evaluate our method against closed-dictionary continuous mapping methods, demonstrating that LatentBKI enables open-dictionary mapping without losing performance and while maintaining the ability to quantify uncertainty. Finally, we evaluate our method on real-world indoor scenes, highlighting the advantage of leveraging open-dictionary segmentation networks for continuous mapping. To summarize, our contributions are:

- 1) Novel mapping algorithm which extends continuous Bayesian Kernel Inference (BKI) to latent spaces.
- Spatial smoothing and uncertainty quantification through conjugate priors in VL maps.
- 3) Demonstration of segmentation and uncertainty quantification in real-world environments.
- 4) Open-source software is available for download at https://github.com/UMich-CURLY/LatentBKI.

# II. RELATED WORK

We review the literature on semantic mapping using continuous probabilistic inference, which creates comprehensive maps with quantifiable uncertainty but is limited to predefined categories. We then examine VL networks and maps, which allow for open-dictionary segmentation at inference time. LatentBKI addresses the challenge of integrating continuous probabilistic mapping with VL networks.

# A. Continuous Semantic Mapping

Robots require advanced levels of scene understanding to plan, including knowledge of the geometry and semantic labels of objects and uncertainty associated with the objects to avoid failure due to mistaken object identity. Often, these approaches use task-dependent object designations as semantic labels [16] such as abstract topological information [21] or material classifications [17], [22]. Robotics research has focused on incorporating segmentation predictions into maps via semantic label fusion [23], [24]. Recent methods aim to quantify uncertainty through Bayesian inference [17], [22] by iteratively fusing semantic estimates projected onto a geometric map. Uncertainty quantification can be used by downstream planning algorithms to identify and circumvent potentially dangerous conditions [17], as well as to plan optimally informative trajectories in a field of robotics known as active perception [18], [19], [25].

Kernel-based inference schemes have had notable success [4], [26] in probabilistic semantic mapping. Bayesian Kernel Inference (BKI), proposed by [20], approximate the spatial influence of points at model selection through the usage of a kernel. BKI is an approximation of Gaussian Processes, which are effective at continuous mapping yet suffer from a cubic computational complexity [14], [27], [28]. Effectively, the kernel defines the shape or distribution of a point, deemed the extended likelihood, and can be applied to create an efficient, closed-form Bayesian update with more complete maps and quantifiable uncertainty [15]. While BKI has been applied effectively to semantic mapping [4], [16], [29], semantic maps are inherently limited to a closed set of pre-specified categories. In contrast, we propose to extend the literature of continuous mapping to the latent space of neural networks through a Gaussian likelihood and conjugate pair, allowing for open-vocabulary inference within the latent space of VL models, without any loss in performance.

# B. Vision Language Mapping

Rapidly improving Large Language Models (LLMs) demonstrating remarkable generalizable capabilities have motivated the advent of vision-language models (VLM) with shared latent space for both images and texts [7], [30], [31]. The pioneering method CLIP successfully represents visual and textual information in the same embedding dimension through contrastive learning [6]. Trained on a large dataset of imagetext pairs, CLIP learns to embed features from visual or textual information in a shared feature space, where similarity is measured by a cosine similarity function [6].

Based on the success of VLMs and their great zero-shot performance, recent robotics research has focused on opendictionary mapping which operates in the latent space of VLMs and can create segmentation predictions from language descriptors [8]. Approaches like LM-Nav [32], CoW [9], NLMap [12] and VLMap [10] have fused VLMs to enable robots to understand and navigate new environments. One common approach in literature to open-dictionary mapping is a volumetric averaging technique [10], [11], [13] where images are processed through a language-driven semantic segmentation network such as LSeg [8], producing 3D points paired with neural features. Points are then incrementally fused within a volumetric map structure as a moving average of the features of points falling within each map cell. At inference time, the latent expectation of each voxel can then be decoded into per-category scores given the language embeddings of a set of categories, thereby enabling open-dictionary queries. While successful, this approach loses the ability to quantify uncertainty and fill in gaps in the map from probabilistic continuous mapping, as discussed previously. Therefore, we propose to extend closed-dictionary continuous semantic mapping to open-dictionary VL maps to obtain quantifiable uncertainty and spatial smoothing.



Fig. 2: This figure demonstrates the overall pipeline of LatentBKI. (a) The input to LatentBKI is 3D points, which can be from LiDAR, RGB-D, or any exteroceptive sensor with 3D input. (b) Points are then processed by an off-the-shelf neural network, which encodes each point into a latent space. (c) By adopting a Gaussian likelihood over the point-wise features, we perform closed-form Bayesian inference on a voxel map where each voxel contains parameters modeling the conjugate prior of the multivariate Gaussian distribution. Additionally, instead of only considering points which fall within a voxel, we consider nearby points weighted through a kernel function. (d) The posterior predictive distribution of each voxel in latent space can then be decoded using the decoder of the neural network, enabling the computation of open-dictionary segmentation predictions with expectation and uncertainty.

#### III. METHOD

We propose a novel method for probabilistic continuous mapping in the feature space of neural networks, which recurrently incorporates predictions from neural networks to learn an expectation and variance. Our mapping framework, which we call LatentBKI, has applications for general deep neural networks and is especially powerful when combined with modern foundation models such as VL models. Compared to previous methods which map in an explicit categorical space, continuous mapping in the feature space allows for opendictionary queries with quantifiable uncertainty. A diagram of our method is shown in Fig. 2, demonstrating the ability of LatentBKI to complete scenes, decode semantic information, and quantify uncertainty in the latent space.

LatentBKI is built on the intuition that neural network features are geometrically continuous and suitable for kernel methods. Interpolation is a common step in modern neural networks to infer features from geometrically adjacent points, used especially in upsampling or deformable operations. Interpolation of point x on feature grid G with height H and width W can be written as:

$$G_x \approx \frac{\sum_{i=1}^H \sum_{j=1}^W w_{ij} G_{ij}}{\sum_i \sum_j w_{ij}}$$
(1)

where i and j are indices of neighboring cells, and weights w are determined by the distance of query point x to neighboring cells. This equation resembles the Nadaraya-Watson kernel estimate of the expected value, written similarly as:

$$\hat{G}_x = \frac{\sum_{i=1}^{N} k(x - x_i) G_i}{\sum_{i=1}^{N} k(x - x_i)}$$
(2)

for a set of N data points. As we will show next, our method produces an expectation equivalent to the Nadaraya-Watson kernel estimate, with the addition of quantifiable uncertainty through conjugate priors.

#### A. Latent Mapping Representation

Our map representation consists of a voxel map with voxels \* located at position  $x_*$  At each time-step our map is provided a set of points  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ , where  $x_i \in \mathbb{R}^3$  is the position of point i and  $y_i \in \mathbb{R}^C$  is the corresponding latent feature of point i. From these points, our goal is to probabilistically update the latent parameters of each voxel,  $y_*$ , to obtain the *posterior*.

In order to accomplish this goal, we first define a Gaussian likelihood over the feature space, such that:  $p(y_i | \mu_i, \Sigma_i) = \mathcal{N}(y_i; \mu_i, \Sigma_i)$ . Since the features originate from a neural network, the likelihood defines an unknown expectation and variance which the point's features are sampled from. Similarly, we can define the points observed within a voxel \* according to the same likelihood distribution. From the likelihood, we can write an expression for the posterior of the latent parameters  $\theta_* = {\mu_*, \Sigma_*}$  of voxel \* using Bayes' rule as:

$$p(\theta_* \mid \boldsymbol{x}_*, \mathcal{D}) \propto p(\mathcal{D} \mid \theta_*, \boldsymbol{x}_*) p(\theta_* \mid \boldsymbol{x}_*).$$
(3)

In order to model the distribution over the parameters  $\theta_*$  of the voxel, which themselves define a multivariate Gaussian distribution, we adopt the conjugate prior of the multivariate Gaussian distribution, the normal-inverse Wishart distribution. The normal-inverse Wishart distribution defines a distribution over the multivariate Gaussian with unknown mean  $\mu_*$  and covariance  $\Sigma_*$  through parameters  $\mu'_*, \Psi_*, \lambda_*, v_*$  as:

$$p(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*) = \mathcal{N}\left(\boldsymbol{\mu}_*; \boldsymbol{\mu}'_*, \frac{\boldsymbol{\Sigma}_*}{\boldsymbol{\lambda}_*}\right) \cdot \mathcal{W}^{-1}(\boldsymbol{\Sigma}; \boldsymbol{\Psi}_*, \boldsymbol{v}_*), \quad (4)$$

where the hyper-parameters represent the prior expectation of the mean  $(\mu'_*)$ , the expectation of the covariance  $(\Psi_*)$ , and the confidence in the mean and covariance estimates  $(\lambda_*, v_*)$ . In our case,  $\lambda_*$  and  $v_*$  are equal and correspond to weighted counts of the total observations.

Although the conjugate prior provides a closed-form solution for updating the multivariate normal distribution parameters for each voxel, it does not consider the spatial locations of points. Intuitively, points that are closer to the centroid of the voxel should have a higher influence, while points that are further from the voxel centroid should have a lower influence. Additionally, only updating the voxels which points fall into can lead to sparse maps, as previously noted. Therefore, we adopt the solution of [20], which defines an extended likelihood distribution that considers the spatial relationship of points to voxels through the use of a kernel function  $k(\boldsymbol{x}_i, \boldsymbol{x}_*)$  as:

$$p(\boldsymbol{y}_i|\boldsymbol{x}_i,\boldsymbol{\theta}_*,\boldsymbol{x}_*) \propto p(\boldsymbol{y}_i|\boldsymbol{\theta}_*)^{k(\boldsymbol{x}_i,\boldsymbol{x}_*)}.$$
 (5)

The only requirements when defining the extended likelihood are that  $k(\boldsymbol{x}, \boldsymbol{x}) = 1 \quad \forall x \text{ and } k(\boldsymbol{x}, \boldsymbol{x}_*) \in [0, 1] \forall (\boldsymbol{x}, \boldsymbol{x}_*)$ . Applied to the previously defined Gaussian likelihood, the extended likelihood can be written as:

$$p(\boldsymbol{y}_i|\boldsymbol{\theta}_*, \boldsymbol{x}_i, \boldsymbol{x}_*) = \mathcal{N}\left(\boldsymbol{y}_i; \boldsymbol{\mu}_*, \frac{\boldsymbol{\Sigma}_*}{k(\boldsymbol{x}_i, \boldsymbol{x}_*)}\right).$$
(6)

Following Semantic BKI, we use a symmetric sparse kernel [33] with kernel length l = 0.5 for direct comparison, where d is the Euclidean distance between the two points:

$$k(d) = \begin{cases} \left[\frac{1}{3}(2 + \cos(2\pi \frac{d}{l})(1 - \frac{d}{l}) + \frac{1}{2\pi}\sin(2\pi \frac{d}{l}))\right], & \text{if } d < l \\ 0. & \text{else} \end{cases}$$
(7)

Substituting the extended likelihood into (3), we can now define a spatial update over the voxel parameters as:

$$p(\theta_*|\boldsymbol{x}_*, \boldsymbol{y}_{1:N}, \boldsymbol{x}_{1:N}) \propto \left[\prod_{i=1}^N p(\boldsymbol{y}_i|\theta_*, \boldsymbol{x}_i, \boldsymbol{x}_*)\right] p(\theta_*|\boldsymbol{x}_*).$$
(8)

Next, we present our map update algorithm, which follows the closed-form solution derived by [20].

#### B. Latent Mapping Update

First, we initialize the confidence over the mean and covariance of each voxel to a non-informative value of  $\lambda_* \approx 0$ . As points are observed, the value of  $\lambda_*$  increases, indicating more confidence in the expected mean and covariance of the voxel. At time-step t, voxels are parameterized by prior  $\mu_*^{t-1}$ ,  $\Psi_*^{t-1}$  and confidence  $\lambda_*^{t-1}$ , with input points  $\mathcal{D}$ .

The influence of the new observations is calculated by:  $\bar{k}_* = \sum_{i=1}^N k(\boldsymbol{x}_*, \boldsymbol{x}_i)$ , where the kernel function measures the influence of point *i* over voxel \*. The confidence in the mean and covariance is updated:  $\lambda_*^t = \lambda_*^{t-1} + \bar{k}_*$ . Input observations are then used to compute the new mean as a running average:  $\bar{\boldsymbol{y}}_* = \sum_{i=1}^N \frac{k(\boldsymbol{x}_*, \boldsymbol{x}_i)}{\bar{k}_*} \boldsymbol{y}_i, \, \boldsymbol{\mu}_*^t = \frac{\lambda_*^{t-1} \boldsymbol{\mu}_*^{t-1} + \bar{\boldsymbol{y}}_* \bar{k}_*}{\lambda_*^t}.$ 

**Remark 1.** We note that the formulation of the new mean resembles the Nadaraya-Watson estimate mentioned in Section III-A, as input features are weighted by the kernel function to obtain a weighted average.

Last, following [20], we update the expected covariance by weighting the covariance of the newly observed points:

$$\bar{E}_* = (\bar{y}_* - \mu_*^{t-1})(\bar{y}_* - \mu_*^{t-1})^T,$$
(9)

$$\bar{S}_* = \sum_{i=1}^{N} k(\boldsymbol{x}_i, \boldsymbol{x}_*) (\boldsymbol{y}_i - \bar{\boldsymbol{y}}_*) (\boldsymbol{y}_i - \bar{\boldsymbol{y}}_*)^T, \qquad (10)$$

$$\Psi^{*t} = \Psi^{*t-1} + \bar{S}_* + \frac{\lambda_*^{t-1}\bar{k}_*}{\lambda_*^t}\bar{E}_*.$$
 (11)

As new points are obtained, the above process is repeated to update the mean, covariance, and confidence level.

## C. Inference

After updating the map, we can compute an expectation and variance for features observed within each voxel. First, the distribution can be marginalized to obtain a posterior predictive solution that defines the probability of observing a feature  $y_*$  at voxel centroid  $x_*$ . The posterior predictive distribution for voxel \* is:

$$p(\boldsymbol{y}_*|\boldsymbol{x}_*) = t_{\lambda_*}\left(\boldsymbol{\mu}_*, \frac{\lambda_*+1}{\lambda_*^2}\Psi_*\right), \quad (12)$$

where  $t_{\lambda_*}$  is the multivariate Student-*t* distribution. The multivariate Student-*t* distribution has an expectation of:  $\mathbb{E}(y_*) = \mu_*$ , when  $\lambda_* > 1$ , and a covariance of:  $\operatorname{Cov}(\boldsymbol{y}_*) = \frac{\lambda_*}{\lambda_*-2} \left(\frac{\lambda_*+1}{\lambda_*^2}\Psi_*\right)$ , for  $\lambda_* > 2$ . However, both the expectation and covariance are within the latent space of the neural network and must be decoded to obtain a meaningful interpretation.

When performing open-dictionary inference, the input is a set of phrases W defining semantic categories, which are processed by a language model to obtain text embeddings  $F_w \in \mathbb{R}^C$  per phrase  $w \in W$ . Specifically, in our experiment, we encode each phrase w as a Clip feature vector  $F_w$  of 512 length:  $F_w = \text{Encoder}(w)$ . Inspired by LSeg, we obtain the categorical prediction  $\hat{w}$  as:

$$\hat{w} = \arg\max_{w} \frac{F_{w}^{\mathsf{T}} \mu_{*}}{\|\mu_{*}\|_{2} \|F_{w}\|_{2}}.$$
(13)

**Remark 2.** We note that while we present the decoding for open-dictionary queries, LatentBKI can be decoded into any format using neural network decoders.

#### D. Uncertainty Quantification

While the map update step described above can propagate uncertainty, the covariance is limited to the latent space of the neural network encoder. Therefore, we propose two methods to quantify uncertainty.

First, following the approach of other neural network uncertainty quantification methods, we propose quantifying uncertainty through sampling. To quantify uncertainty through sampling, we sample many realizations of the voxel feature  $y_*$ , which we decode through the neural network decoder. Then, we compute the variance of the predictions in the decoded space. While this approach is accurate, it requires extra computation, which we propose to avoid.

Based on a common approach of information quantification in optimal experimental design [25], we propose to consolidate the covariance matrices as a single scalar value  $U_*$  through p-optimality [34]. Although p-optimality leads to many solutions [35], in this work we calculate and compare E-optimality, which selects the maximum eigen-value of the covariance matrix, and D-optimality, which calculates the volume of the covariance hyper-ellipsoid. For eigen-values  $\lambda$  of the covariance matrix, which we note are the diagonals of an uncorrelated covariance matrix, E-optimality criterion is computed as:  $U_* = \max(\lambda)$ , and D-optimality criterion is computed as:

$$U_* = \exp\left(\frac{\sum_{i=1}^C \log\left(\boldsymbol{\lambda}_i\right)}{C}\right). \tag{14}$$

Experimentally, we find that E-optimality is highly correlated with the sampling-based uncertainty and is quick to compute. See Section IV-C for detailed experiments

#### E. Feature Compression

Due to the large latent dimension of VLM's, we propose to make two approximations to reduce computation and memory complexity. First, we approximate the covariance matrix with only the diagonal elements, significantly reducing complexity at the cost of cross-correlation terms. Diagonal covariances are common in the feature space and are used in variational autoencoders (VAEs). Second, we use PCA to reduce the latent dimension of encoded features from a latent dimension of 512 to 64 from VLM's before fusing into our map. PCA is an unsupervised learning algorithm that maximizes information preserved during compression and uses an affine transformation to upscale the compressed features back to the original dimension. Since the transformation is affine, we can compute the full dimensional expectation by passing the compressed expectation through the PCA upsampling. For the uncertainty, we compute E and D optimality in the reduced latent space and obtain full dimensional samples for the sampling technique through PCA upsampling. Overall, we found that PCA is generalizable to new scenes with minimal performance loss from compression.

### IV. RESULTS AND DISCUSSION

We quantitatively and qualitatively show that LatentBKI effectively extends the continuous probabilistic mapping literature to neural network latent spaces, bringing quantifiable uncertainty and spatial smoothing to VL maps. First, we compare LatentBKI against closed-dictionary continuous mapping to verify that operations in the latent space of neural networks do not affect mapping performance. Next, we compare LatentBKI against VLMap, which performs latent space mapping but does not leverage spatial information or quantify uncertainty. Third, we study the correlation of our uncertainty quantification with segmentation errors and the effect of spatial smoothing. Finally, we conduct real-world experiments to demonstrate the ability of our map to transfer to real-world open-dictionary scenarios due to the strong generalization capabilities of large VL networks.

Quantitative results are obtained on popular outdoor and indoor datasets. For the outdoor results, we compare on the validation set of the Semantic KITTI dataset [36] using the Sparse Point-Voxel Convolution Neural Network (SPVCNN) [37] for point cloud semantic prediction. We choose the validation set because it has publicly available ground truth, and 4,070 frames. For the indoor comparison, we evaluate methods on all eight scenes of the Matterport3D (MP3D) [38] dataset with an open-dictionary image semantic segmentation network, Language-driven Semantic Segmentation (LSeg) [8]. Since the latent space of LSeg has a large dimension of 512, we use PCA to down-sample features for the map update to a size of 64.

### A. Comparison against categorical space mapping method

First, we compare LatentBKI against the closed-dictionary semantic mapping method defined in [16], which leverages BKI with a categorical likelihood to update the map. Specifically, we compare against ConvBKI [4] using a single untrained spherical kernel for direct comparison. Our goal of this study is to verify that LatentBKI can generalize BKI into the latent space of neural networks without any significant changes in performance. Note that in this experiment, similar quantitative results indicate successful application of BKI to the latent space without any loss of functionality.

We apply the same configuration to both mapping algorithms to ensure comparable results. Each algorithm uses a voxel resolution of 0.1 meters, a sparse kernel with a kernel length of 0.5 meters, and a filter size of 3, determining how many neighboring voxels should be updated for a single-point observation along a single axis. Since both methods compare spatial smoothing, we provide 80% of the points as input and evaluate semantic predictions over the mean intersection over union (mIoU) and accuracy metrics on the remaining 20% of the points.

As shown in Table I, LatentBKI performs similarly to ConvBKI over indoor and outdoor datasets without any decrease in performance. These results verify that our approach generalizes BKI to the latent space of networks, enabling open-dictionary probabilistic mapping, successfully. While LatentBKI results in a marginal improvement in quantitative performance on the outdoor data, the slight decrease in indoor data is due to the dimensionality reduction applied by PCA on the input to LatentBKI. Next, we compare LatentBKI with a popular latent mapping algorithm, VLMap.

### B. Latent Mapping Comparison

We compare LatentBKI with a similar open-dictionary mapping method VLMap [10], which also updates voxels through a weighted averaging approach. We choose to compare specifically with VLMap since weighted averaging is a popular technique for open-dictionary mapping [10], [11], [13], and the volumetric representation of VLMap allows for a direct and conclusive comparison. Since our approach generalizes VLMap to include a spatial kernel and quantifiable uncertainty, we compare the results with (k = 3) and without (k = 1)

TABLE I: Comparison against closed-dictionary BKI mapping.

Data	Method	Acc. (%)	mIoU (%)	Queries
	Segmentation	59.14	14.64	N/A
Indoor	ConvBKI (Single)	61.49	16.69	Fixed
	LatentBKI	60.44	16.15	Open
	Segmentation	89.60	58.54	N/A
Outdoor	ConvBKI (Single)	90.02	61.26	Fixed
	LatentBKI	90.02	61.54	Open

TABLE II: Latent mapping comparison on MP3D.

Method	Acc. $(\%)$	mloU (%)	Uncertainty
Segmentation [8]	53.24	12.59	N/A
Heuristic	51.63	11.60	No
VLMap [10]	53.84	12.53	No
LatentBKI $(k = 1)$	55.57	14.01	Yes
LatentBKI $(k = 3)$	55.86	14.18	Yes

spatial smoothing, where k is the number of neighboring voxels along each dimension an observation can influence. To demonstrate the benefits of VLMap, we also implement a heuristic baseline that stores the feature of the most recent coinciding observation within each voxel.

We compare each method on the MP3D dataset [38] following the same experimental setup as VLMap, including a resolution of 0.05 m to account for the fine-resolution indoor environment. Additionally, following the setup of VLMap we discard pixels with extreme depths < 0.1 m or > 6 m, discard points outside of the scene range, and downsample input points to the same set of 1% of input pixels. By following the same downsampling heuristics as VLMap in our evaluation, we isolate the effect of the sparse kernel function and spatial smoothing used by our method to weight input points compared to the depth-wise weighting scheme employed by VLMap [13], [39].

The results of our experiments in Table II indicate that both LatentBKI and VLMap outperform the heuristic baseline, demonstrating the benefit of the weighted average approach. LatentBKI outperforms VLMap in both accuracy and mean IoU, which we attribute to the sparse kernel and spatial smoothing. We note that our method is a probabilistic generalization of VLMap with a spatial kernel and quantifiable uncertainty, however these benefits also increase computational complexity linearly with the size of the spatial kernel. Although the computational complexity of our method is greater, our implementation is more efficient than VLMap due to vectorization and the use of the GPU, requiring 122.05 ms to update 100,000 points compared to 4,325.74 ms for VLMap to update the same number of points.

## C. Ablation Studies

Two benefits of LatentBKI are the spatial smoothing effect of continuous mapping and the ability to quantify the temporal uncertainty of neural network predictions. In this section, we study the quantitative improvement from different kernel sizes, as well as compare the sampling and P-Optimality methods for quantifying uncertainty.

**Spatial Smoothing:** In real-world applications, data is often sparse due to sensors such as LiDAR or sparse stereo matching algorithms. BKI provides a probabilistic technique to create more complete maps from sparse spatial data by leveraging the spatial smoothing effect of kernels. In this experiment, we compare different kernel sizes (k) and their ability to complete the map from sparse data.

All kernels are compared on the same scene of MP3D, where data is downsampled temporally to incorporate only one in 3 frames and at the image level to use a randomly sampled set of pixels from each image. Fig. 3 demonstrates plots of the segmentation performance of different kernel sizes and varying



Fig. 3: Effect of spatial smoothing with varying levels of image sparsity. Spatial smoothing, indicated by the filter size k, is most effective for sparse images. The original image has a resolution of 720 by 1080 pixels.

sparsity levels. At extreme sparsity levels, spatial smoothing of points benefits the map completeness. As the input becomes more dense, the effect of the kernel is diminished.

**Uncertainty Quantification:** To evaluate the ability of LatentBKI to quantify uncertainty meaningfully, we quantitatively and qualitatively compare uncertainty quantification on the MP3D dataset using LSeg as the encoder. We construct a map using LatentBKI, then compare uncertainty quantified using D-optimality, E-optimality, and sampling as described in Section III-D. Whereas the sampling-based method is commonly used, it is computationally expensive compared to the E-optimality and D-optimality-based techniques with a run-time of 5, 661 ms for 10,000 query voxels compared to 2.3 ms for the optimality approaches to compute.

To quantitatively compare each uncertainty quantification method, we create sparsification plots [40] identifying the correlation between uncertainty and prediction error, shown in Fig. 4. To create the sparsification plots, we sort points in a test set by the predicted uncertainty. Next, we separate the sorted points into bins and iteratively remove the most uncertain bin. If the uncertainty is properly calibrated, we expect to see an increase in the accuracy as uncertain points are removed. As seen in Fig. 4, both the accuracy and mIoU metrics are correlated with all three methods, especially sampling and Eoptimality. In addition to a strong correlation between latent uncertainty and error in the decoded predictions, E-optimality benefits from efficient computation.

We also qualitatively compare uncertainty quantification between sampling and E-optimality, shown in Fig. 5. We observe that the most uncertain voxels are typically located at the edges of rooms or at objects that are difficult for the VL network to identify due to ambiguity or poorly captured images. Similar to the sparsification plots, we find that E-optimality closely aligns



Fig. 4: Sparsification plot of segmentation performance compared to quantified uncertainty. As uncertain points are removed, a well calibrated uncertainty should cause the segmentation performance to increase.



Fig. 5: Uncertainty maps for 5LpN3gDmAk7 MP3D sequence. (a) Covered house mesh in the sequence. (b) Categorical variance map by sampling from distribution. (c) Variance map by using E-optimality in latent space. (d) Variance map by using D-optimality in latent space.

with the uncertainty estimated through sampling, indicating that E-optimality is an effective approximation for the latent uncertainty.

## D. Real-World Experiment

To show that LatentBKI can generalize to real-world settings, we use an iPad with a 3D recording software, Record3D, to collect RGB-D data and camera poses of indoor scenes for mapping. We process the images with LSeg, and create a map of a real-world apartment using LatentBKI, shown in Figure 6.

In Fig. 6, we demonstrate how Latent-BKI enables openvocabulary queries which are more suitable for complex indoor environments. In this figure, we query arbitrary words in the map and portray a heat map of the voxels corresponding to the query word. While results were compared on a closed set of segmentation categories, our method enables language-based inference with quantifiable uncertainty. This is especially important because indoor environments can contain infinitely many categories of objects that cannot be captured adequately with a pre-specified set of objects.



(c) Query "bed"

Fig. 6: Open vocabulary query task results. Query results are shown in heat map, brighter colors indicates higher values.



Fig. 7: The first column (a) shows RGB input across different frames containing "bed". Column two (b) shows the Lseg network semantic prediction result, which gives inconsistent wrong semantic prediction across frames. Later two columns, (c) shows the heatmap of the query result of "bed" and (d) show the uncertainty. Our method shows a consistently high probability in the same area for the "bed" while maintaining the knowledge that the observations on the "bed" area are noise by showing high uncertainty.

An additional benefit of LatentBKI is the ability to quantify uncertainty, which we demonstrate in Fig. 7. The input segmentation network has difficulty identifying a vertically placed mattress, producing inconsistent embeddings across different views. As a result, this region of the map exhibits high variance. Although the network prediction is noisy, LatentBKI can generate consistent query results for "bed" while acknowledging the high uncertainty from the network in that area.

# V. CONCLUSION

We introduced LatentBKI, a novel method for probabilistically updating a voxel map where each voxel stores a latent descriptor in the embedding space of foundation models with quantifiable uncertainty. LatentBKI extends the classical literature of continuous semantic mapping to open-dictionary mapping, enabling language-based queries while maintaining quantifiable uncertainty. Language-based queries can handle the complexities posed by real-world robotic applications that may contain detailed environments and require human interaction.

While LatentBKI demonstrated success in open-dictionary environments through a Gaussian likelihood, there are several avenues for future work. First, LatentBKI does not consider the unique geometry of objects and can therefore be combined with architectures such as ConvBKI, which learns per-category kernels, or the high-quality 3D Gaussian Splatting [41] novel view synthesis method which represents the environment using 3D ellipsoids, similar to the kernel structure we leverage for continuous mapping. Additionally, inspired by the recent work on open-dictionary radiance fields [42], we believe that the segment anything model [5] may be useful when identifying the boundaries of objects. Last, the uncertainty produced by LatentBKI may be used for planning tasks such as active perception by quantifying expected information gain [18], [19], [25].

#### ACKNOWLEDGMENT

This work was partly supported by the DARPA TIAMAT project. M. Ghaffari thanks Dr. Alvaro Velasquez for the encouragement and support.

#### References

- S. Casas, A. Sadat, and R. Urtasun, "MP3: A Unified Model to Map, Perceive, Predict and Plan," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2021, pp. 14403–14412.
- [2] H. L. Chiang, A. Faust, M. Fiser, and A. Francis, "Learning Navigation Behaviors End-to-End With AutoRL," *IEEE Robot. Autom. Letter.*, vol. 4, no. 2, pp. 2007–2014, 2019.
- [3] H. Moravec and A. Elfes, "High resolution maps from wide angle sonar," in *Proc. IEEE Int. Conf. Robot. and Automation*, vol. 2, 1985, pp. 116– 121.
- [4] J. Wilson, Y. Fu, A. Zhang, J. Song, A. Capodieci, P. Jayakumar, K. Barton, and M. Ghaffari, "Convolutional Bayesian Kernel Inference for 3D Semantic Mapping," in *Proc. IEEE Int. Conf. Robot. and Automation*, 2023, pp. 8364–8370.
- [5] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollar, and R. Girshick, "Segment anything," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2023, pp. 4015–4026.
- [6] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning Transferable Visual Models From Natural Language Supervision," in *Proc. Int. Conf. Machine Learning*, ser. J. Mach. Learning Res., vol. 139, 2021, pp. 8748–8763.
- [7] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds *et al.*, "Flamingo: a visual language model for few-shot learning," *Proc. Advances Neural Inform. Process. Syst. Conf.*, vol. 35, pp. 23716–23736, 2022.
- [8] B. Li, K. Q. Weinberger, S. Belongie, V. Koltun, and R. Ranftl, "Language-driven Semantic Segmentation," in *Proc. Int. Conf. Learning Representations*, 2022.
- [9] S. Y. Gadre, M. Wortsman, G. Ilharco, L. Schmidt, and S. Song, "CoWs on Pasture: Baselines and Benchmarks for Language-Driven Zero-Shot Object Navigation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2023, pp. 23171–23181.
- [10] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual Language Maps for Robot Navigation," in *Proc. IEEE Int. Conf. Robot. and Automation*, 2023, pp. 10608–10615.
- [11] ——, "Audio Visual Language Maps for Robot Navigation," in *Exper. Robot.*, 2024, pp. 105–117.
- [12] B. Chen, F. Xia, B. Ichter, K. Rao, K. Gopalakrishnan, M. S. Ryoo, A. Stone, and D. Kappler, "Open-vocabulary Queryable Scene Representations for Real World Planning," in *Proc. IEEE Int. Conf. Robot.* and Automation, 2023, pp. 11 509–11 522.
- [13] K. Jatavallabhula, A. Kuwajerwala, Q. Gu, M. Omama, T. Chen, S. Li, G. Iyer, S. Saryazdi, N. Keetha, A. Tewari, J. Tenenbaum, C. de Melo, M. Krishna, L. Paull, F. Shkurti, and A. Torralba, "ConceptFusion: Open-set Multimodal 3D Mapping," in *Robotics. Sci. Sys.*, 2023.
- [14] S. T. O'Callaghan and F. T. Ramos, "Gaussian process occupancy maps," *Int. J. Robot. Res.*, vol. 31, no. 1, pp. 42–62, 2012.
- [15] K. Doherty, T. Shan, J. Wang, and B. Englot, "Learning-Aided 3-D Occupancy Mapping with Bayesian Generalized Kernel Inference," *IEEE Trans. Robot.*, vol. 35, no. 4, pp. 953–966, 2019.
- [16] L. Gan, R. Zhang, J. W. Grizzle, R. M. Eustice, and M. Ghaffari, "Bayesian Spatial Kernel Smoothing for Scalable Dense Semantic Mapping," *IEEE Robot. Autom. Letter.*, vol. 5, no. 2, pp. 790–797, 2020.
- [17] P. Ewen, H. Chen, Y. Chen, A. Li, A. Bagali, G. Gunjal, and R. Vasudevan, "You've Got to Feel It To Believe It: Multi-Modal Bayesian Inference for Semantic and Property Prediction," in *Robotics. Sci. Sys.*, 2024.
- [18] G. Georgakis, B. Bucher, K. Schmeckpeper, S. Singh, and K. Daniilidis, "Learning to Map for Active Semantic Goal Navigation," in *Proc. Int. Conf. Learning Representations*, 2022.
- [19] G. Georgakis, B. Bucher, A. Arapin, K. Schmeckpeper, N. Matni, and K. Daniilidis, "Uncertainty-driven Planner for Exploration and Navigation," in *Proc. IEEE Int. Conf. Robot. and Automation*, 2022, pp. 11 295–11 302.
- [20] W. R. Vega-Brown, M. Doniec, and N. G. Roy, "Nonparametric Bayesian inference on multivariate exponential families," in *Proc. Advances Neural Inform. Process. Syst. Conf.*, vol. 27, 2014.
- [21] B. Kuipers and Y.-T. Byun, "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations," *Robot. and Auton. Syst.*, vol. 8, no. 1-2, pp. 47–63, 1991.

- [22] P. Ewen, A. Li, Y. Chen, S. Hong, and R. Vasudevan, "These maps are made for walking: Real-time terrain property estimation for mobile robots," *IEEE Robot. Autom. Letter.*, vol. 7, no. 3, pp. 7083–7090, 2022.
  [23] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "Semanticfu-
- [23] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "Semanticfusion: Dense 3d semantic mapping with convolutional neural networks," in *Proc. IEEE Int. Conf. Robot. and Automation*. IEEE, 2017, pp. 4628–4635.
- [24] N. Sünderhauf, T. T. Pham, Y. Latif, M. Milford, and I. Reid, "Meaningful maps with object-oriented semantic mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, 2017, pp. 5079–5085.
- [25] J. A. Placed, J. Strader, H. Carrillo, N. Atanasov, V. Indelman, L. Carlone, and J. A. Castellanos, "A Survey on Active Simultaneous Localization and Mapping: State of the Art and New Frontiers," *IEEE Trans. Robot.*, vol. 39, pp. 1686–1705, 2022.
- [26] L. Gan, Y. Kim, J. W. Grizzle, J. M. Walls, A. Kim, R. M. Eustice, and M. Ghaffari, "Multitask learning for scalable and dense multilayer Bayesian map inference," *IEEE Trans. Robot.*, vol. 39, no. 1, pp. 699– 717, 2022.
- [27] M. G. Jadidi, L. Gan, S. A. Parkison, J. Li, and R. M. Eustice, "Gaussian Processes Semantic Map Representation," *ArXiv*, vol. abs/1707.01532, 2017.
- [28] J. Wang and B. Englot, "Fast, accurate gaussian process occupancy maps via test-data octrees and nested Bayesian fusion," in *Proc. IEEE Int. Conf. Robot. and Automation*, 2016, pp. 1003–1010.
- [29] J. Wilson, Y. Fu, J. Friesen, P. Ewen, A. Capodieci, P. Jayakumar, K. Barton, and M. Ghaffari, "ConvBKI: Real-Time Probabilistic Semantic Mapping Network with Quantifiable Uncertainty," *IEEE Trans. Robot.*, vol. 40, pp. 4648–4667, 2024.
- [30] L. Yao, R. Huang, L. Hou, G. Lu, M. Niu, H. Xu, X. Liang, Z. Li, X. Jiang, and C. Xu, "FILIP: Fine-grained interactive language-image pre-training," in *Proc. Int. Conf. Learning Representations*, 2022.
- [31] Y. Li, F. Liang, L. Zhao, Y. Cui, W. Ouyang, J. Shao, F. Yu, and J. Yan, "Supervision Exists Everywhere: A Data Efficient Contrastive Language-Image Pre-training Paradigm," in *Proc. Int. Conf. Learning Representations*, 2022.
- [32] D. Shah, B. Osiński, b. ichter, and S. Levine, "LM-Nav: Robotic Navigation with Large Pre-Trained Models of Language, Vision, and Action," in *Conf. Robot. Learning.*, vol. 205, 2023, pp. 492–504.
- [33] A. Melkumyan and F. Ramos, "A Sparse Covariance Function for Exact Gaussian Process Inference in Large Datasets," in *Proc. Int. Joint Conf. Artif. Intell.*, 2009, p. 1936–1942.
- [34] J. Kiefer, "General equivalence theory for optimum designs (approximate theory)," *The annals of Statistics*, pp. 849–879, 1974.
- [35] J. A. Placed and J. A. Castellanos, "A General Relationship between Optimality Criteria and Connectivity Indices for Active Graph-SLAM," *IEEE Robot. Autom. Letter.*, vol. 8, no. 2, pp. 816–823, 2022.
- [36] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019.
- [37] H. Tang, Z. Liu, S. Zhao, Y. Lin, J. Lin, H. Wang, and S. Han, "Searching Efficient 3D Architectures with Sparse Point-Voxel Convolution," in *Proc. European Conf. Comput. Vis.*, 2020, p. 685–702.
- [38] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niebner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3D: Learning from RGB-D Data in Indoor Environments," in *Proc. IEEE Int. Conf. 3D Vis.*, 2017, pp. 667–676.
- [39] M. Keller, D. Lefloch, M. Lambers, S. Izadi, T. Weyrich, and A. Kolb, "Real-Time 3D Reconstruction in Dynamic Scenes Using Point-Based Fusion," in *Proc. IEEE Int. Conf. 3D Vis.*, 2013, pp. 1–8.
- [40] E. Ilg, Ö. Çiçek, S. Galesso, A. Klein, O. Makansi, F. Hutter, and T. Brox, "Uncertainty Estimates and Multi-hypotheses Networks for Optical Flow," in *Proc. European Conf. Comput. Vis.*, 2018, pp. 677– 693.
- [41] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3D Gaussian Splatting for Real-Time Radiance Field Rendering," ACM Trans. Graphics., vol. 42, no. 4, 2023.
- [42] M. Qin, W. Li, J. Zhou, H. Wang, and H. Pfister, "LangSplat: 3D Language Gaussian Splatting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, June 2024, pp. 20051–20060.