A Learned Approach to Adaptive Sampling for Seabed Identification with Autonomous Vehicles

John Lipor
Department of Electrical and Computer Engineering
Portland State University
Email: lipor@pdx.edu

***Abstract*—We propose a sampling approach for guiding an autonomous underwater vehicle to adaptively collect ambient acoustic sound, with the goal of partitioning the seabed according to its geoacoustic properties. Existing approaches to adaptive sampling are based on heuristic policies that aim to balance exploration of the entire region with refinement of the existing partition estimates. We utilize behavioral cloning, which trains a neural network to imitate an expert policy from simulated data, training the policy to maximize the area under the curve of F1 score versus distance traveled. Results on synthetic and real-world sediment data show our approach outperforms existing methods in terms of sampling efficiency.**

***Index Terms*—adaptive sampling, ambient noise, autonomous vehicle, behavioral cloning, Gaussian processes, geoacoustic inversion, reinforcement learning**

I. Introduction

Obtaining an accurate understanding of the geoacoustic properties of the seabed is a topic of interest to scientists and engineers, with applications in sonar performance prediction and the impact of ocean noise on marine life [1]. To obtain estimates of these properties over large regions of interest, recent research has considered the use of autonomous underwater vehicles (AUVs) tasked with partitioning the ocean floor according to similar seabed types [2]. An example of one such partitioning is shown in Fig. 1(a), where each color corresponds to a distinct sediment type defined in the High Frequency Environmental Acoustics (HFEVA) dataset [3]. Since AUVs have strict power requirements, they cannot actively transmit acoustic signals and therefore must utilize the ambient sound in the ocean to perform estimation. Further, to obtain high spatial resolution, these vehicles can benefit from determining their sampling paths adaptively, i.e., choosing sample locations based on all previous locations and measurements.

Existing approaches to adaptive sampling utilize a variety of heuristics to balance *exploration* (discovering new connected components of like seabed type) with *exploitation* (refining the boundary between discovered components). The former is achieved by sampling locations of high uncertainty, while the latter involves sampling locations near the current boundary estimate. In [2], the authors propose a means of balancing these goals by sampling locations that are likely to have the maximal impact on uncertainty reduction. This is achieved by a lookahead method that estimates the possible reduction in uncertainty by bounding the range of values that measurements from a given location can fall in. While this approach obtains state-of-the-art performance on seabed identification tasks, the bounding technique used is loose, resulting in inaccurate estimates of uncertainty reduction. Further, uncertainty reduction is still a heuristic that is tied to estimation accuracy but does not capture it directly.

J. Lipor was supported by Defense Advanced Research Projects Agency YFA award 454 D22AP00135 and National Science Foundation award CAREER CIF-2046175.

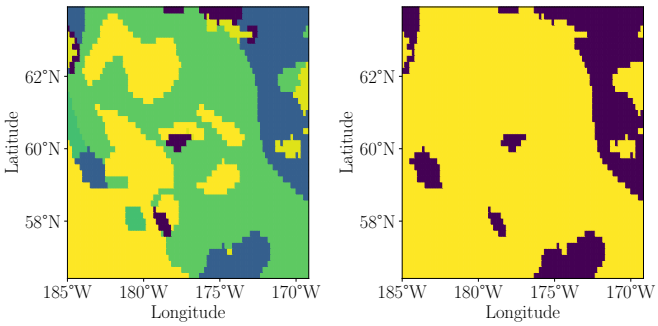


Fig. 1. Example seabed partitioning according to HFEVA sediment types in the northern Pacific Ocean. (a) Colors depict distinct seabed types. The goal is to discover all light/dark blue regions, which differ significantly from the background seabed types clay (yellow) and medium silt (green). (b) Conversion of (a) to level set estimation problem, where color indicates sublevel set (yellow) and superlevel set (blue).

In this work, we present a reinforcement learning approach to adaptive sampling that aims to directly maximize the area under the curve (AUC) of F1 score versus distance traveled. We demonstrate how an expert policy can be formed using simulated data, and then train a neural network to imitate this expert, predicting the AUC after sampling at a given location. Our policy then samples the location with the highest predicted AUC, resulting in significant performance improvements over existing methods, even on fields that differ significantly from the training data. We demonstrate these benefits on both synthetic and real-world sediment data.

II. Problem Formulation & Related Work

Consider an AUV equipped with an array of M receivers used to capture ambient sound and traveling over a region of interest (domain) $\mathcal{D} \subset \mathbb{R}^2$. For a reference location $x_0 \in \mathcal{D}$, our goal is to discover all locations whose seabed type is significantly different from the reference location while mini-

mizing the distance traveled. To accomplish this, time-series pressure recordings are Fourier transformed to obtain single-frequency snapshots $z_1^{(t)}, \dots, z_L^{(t)} \in \mathbb{C}^M$, which, for a given location $x_t \in \mathcal{D}$, are assumed to be drawn from a circularly-symmetric complex normal distribution with zero mean and covariance Σ_t. The similarity between x_0 and x_t is obtained by exponentiating the Jeffreys divergence (symmetrized Kullback-Leibler divergence) between the estimated distributions at each location. Let $\hat{\Sigma}_0$ and $\hat{\Sigma}_t$ be the sample covariance matrices obtained from L snapshots collected at each of the locations x_0 and x_t, respectively. A noisy estimate of the similarity between these locations is then obtained as

$$s_t = \exp\left(-J\left(\hat{\Sigma}_0 || \hat{\Sigma}_t\right)/\ell^2\right), \tag{1}$$

where $J(\cdot||\cdot)$ denotes the Jeffreys divergence and $\ell > 0$ is a tuning parameter used to control the scale of the similarities. Our goal is to estimate the sublevel set of locations that are dissimilar to the reference location

$$\mathcal{L} = \{x \in \mathcal{D} : s(x_0, x) \leq \tau\}, \tag{2}$$

where $\tau > 0$ is the similarity threshold and $s(x_0, x)$ is the similarity between the locations according to the true (unknown) distributions. Since the major cost to sampling with autonomous vehicles is due to the distance traveled between sampling locations, we wish to obtain an accurate estimate of $\mathcal{L}$ while minimizing the distance traveled.

Partitioning $\mathcal{D}$ into its sublevel and superlevel sets is a problem known as *level set estimation* (LSE), and adaptive sampling for LSE has been a topic of extensive study [4]–[7]. An example partitioning of a region into sublevel and superlevel sets is shown in Fig. 1(b), where the reference location is the bottom left corner and the sublevel set is depicted in yellow. To perform LSE, we estimate the similarities across the entire domain $\mathcal{D}$ using Gaussian process (GP) regression [8]. At time t, we have visited a set of locations $x_1, \dots, x_t$ and obtained the corresponding similarity estimates $s_1, \dots, s_t$. Let $k(x, x')$ be the kernel function that defines the similarity between locations x and x', and define $K_t \in \mathbb{R}^{t \times t}$ to be the matrix whose i, jth entry is $k(x_i, x_j)$. Let $k_{x,t} = [k(x, x_1), \dots, k(x, x_t)]$, and $y_t = [s_1, \dots, s_t]$. Under the assumption that the similarities are corrupted by Gaussian noise with zero mean and variance γ, we estimate the similarity between the reference location and any location $x \in \mathcal{D}$ via the prediction

$$\mu_t(x) = k_{x,t}^T \left(K_t + \gamma I\right)^{-1} y_t. \tag{3}$$

For each estimate, the F1 score can be computed by comparing $\hat{\mathcal{L}}_t = \{x \in \mathcal{D} : \mu_t(x) \leq \tau\}$ with the true sublevel set $\mathcal{L}$.

In addition to the predicted mean (3), the GP model also provides a posterior variance $\sigma_t^2(x)$, which provides the uncertainty at each location and can be computed as

$$\sigma_t^2(x) = k(x, x) - k_{x,t}^T \left(K_t + \gamma I\right)^{-1} k_{x,t}. \tag{4}$$

When samples are collected sequentially, the above matrix inverse can be updated efficiently following [9], [10]. The standard approach to GP-LSE is to utilize the confidence bounds derived in [4] to form the certain sets

$$L_t = \{x \in \mathcal{D} : \mu_t(x) + \eta\sigma_t(x) \leq \tau\} \tag{5}$$

$$H_t = \{x \in \mathcal{D} : \mu_t(x) - \eta\sigma_t(x) \geq \tau\} \tag{6}$$

whose level set membership is correct with high probability for some $\eta > 0$. The remaining points belong to the *uncertain set*

$$U_t = \mathcal{D} \setminus (L_t \cup H_t), \tag{7}$$

which is the set of points whose level set membership cannot be determined at time t.

Various approaches aim to reduce the cardinality of the uncertain set by sampling points near the level set boundary, of high posterior variance, or some combination of the two [4], [11]. Since the posterior variance at location x does not depend on the measurement value, the authors of [5] show that sampling locations based on the reduction in posterior variance leads to strong LSE performance. Extending this idea further, the lookahead uncertain set reduction (LUSR) algorithm [2] estimates the reduction in cardinality of U_t directly. While this results in further performance benefits, these methods all treat the distance traveled while sampling myopically, either normalizing by the distance from the current location [5] or by selecting from among the nearest neighbors in U_t [2]. In [12], the authors show that treating distance traveled nonmyopically allows even very simple algorithms to outperform GP approaches to LSE, though this approach only applies to simply-connected sublevel sets. In [13], the authors formulate distance-penalized LSE as a stochastic shortest path problem that can be efficiently solved via dynamic programming for a very simple sublevel set model. In this work, our goal is to utilize reinforcement learning to allow for nonmyopic distance penalization in realistic environments.

Training RL agents can be a difficult task due to various decisions related to the design of the state space, reward function, as well as the various hyperparameters required by RL algorithms. One simple approach to RL is known as *behavioral cloning* (BC), in which the goal is to train a neural network to imitate the behavior of an expert agent [14], [15]. While simple to train, BC agents often fail when encountering states not seen in the training data, and the sub-optimal approximations of expert actions can lead to a shift in the distribution of the encountered states, resulting in poor real-world performance. In our setting, we have two advantages that allow for BC to obtain strong performance. First, the GP model is inherently smooth, i.e., sampling nearby locations has a similar impact on the uncertain set, making the impact of sampling easy to approximate with a neural network. Second, we can incorporate prior knowledge into the state space via the predicted reduction in variance. Since the posterior variance update (4) does not depend on the measurement value itself, the reduction in variance can be included as a feature when learning to mimic expert behavior. This latter benefit distinguishes our application of BC from existing approaches and likely results in the strong

performance achieved by our approach, even on data whose state distribution differs significantly from the training data.

III. Proposed Sampling Approach

To apply BC to our setting, we first require an expert policy that the RL agent will learn to imitate. We compare our sublevel set estimate with the ground truth using F1 score, which is a measure of accuracy that is sensitive to class imbalance. We then score a sampling policy using the AUC of F1 score versus distance traveled. Since an ideal policy will obtain an accurate estimate without traveling a large distance, we aim to maximize this AUC.

To derive an expert policy, we use an approach in the spirit of rollout [16], where the value of sampling a given location is based on the immediate reward of sampling that location plus the estimated rewards after following some sub-optimal base policy. Let the distance traveled up to time t be d_t, and define $\alpha(x_t)$ to be the marginal AUC over the interval $[d_{t-1}, d_t]$. Our goal is to maximize $\sum_{t=1}^{T} \alpha(x_t)$, where T is the unknown stopping time at which a distance $d_{\max}$ has been traveled and $d_0 = 0$. Since maximizing the AUC directly via dynamic programming is computationally intractable, we follow an approximation approach. At time t, let $V_t(x_t) = \alpha(x_t) + \sum_{i=t+1}^{T} \alpha(x_i)$ be the remaining AUC after sampling location x and then following the base policy to select locations $x_{t+1}, \ldots, x_T$. While calculating $V_t(x_t)$ is computationally feasible, it still may incur a large computational cost when t is small. To overcome this issue, we use a limited lookahead approach with terminal cost approximation. In particular, we sample the location x, follow the base policy for Δ steps, and then estimate the remaining AUC assuming no further accuracy improvements. The resulting estimated AUC is then

$$\hat{V}_t(x_t) = \alpha(x_t) + \sum_{i=t+1}^{t+\Delta+1} \alpha(x_i) + (d_{\max} - d_{t+\Delta+1})\alpha(x_{t+\Delta+1}). \tag{8}$$

Eq. (8) can be broken into three terms: (1) the immediate improvement in AUC after sampling location x, (2) the AUC after sampling Δ steps according to the base policy, and (3) the approximation of the AUC after sampling until traveling a distance $d_{\max}$. At each step, the expert policy chooses the location $x_t \in U_t$ that maximizes $\hat{V}_t(x_t)$. Pseudocode for this policy is given in Alg. 1. Note that computing $\alpha(x_t)$ requires knowledge of the true sublevel set, and hence this expert policy cannot be applied in practice. However, by simulating numerous sublevel set examples, we can obtain millions of $\left(x_t, \hat{V}_t(x_t)\right)$ pairs that can be used to train a neural network to mimic the expert's behavior.

Having developed an expert policy above, our second step is to learn a policy that mimics expert behavior but only utilizes information available to the AUV in real time. In particular, we wish to train a neural network to approximate $\hat{V}_t(x_t)$ for any potential sampling location $x_t \in U_t$. For each $\left(x_t, \hat{V}_t(x_t)\right)$ pair computed by the expert policy, we form the feature/example tensor consisting of a three-channel image.

Algorithm 1 Expert LSE policy used for behavioral cloning.

Input: previous sample locations $x_1, \ldots, x_{t-1}$ and similarities $s_1, \ldots, s_{t-1}$

Output: next sample location x_t^*

compute $\mu_t(x)$ and $\sigma_t^2(x)$ according to (3) and (4)
compute uncertain set U_t according to (7)
for $x_t \in \mathcal{U}_t$ **do**
    compute $\hat{V}_t(x_t)$ according to (8)
end for
$x_t^* = \arg\max_{x_t \in U_t} \hat{V}_t(x_t)$

The first channel is the absolute distance between the current estimate $\mu_t(x)$ and the threshold τ. This feature gives an indication of points that lie near the level set boundary and therefore whose level set membership is difficult to determine. The second channel is the current posterior variance $\sigma_t^2(x)$, which provides the degree of uncertainty about the level set estimates. The third channel is the reduction in uncertainty after sampling location x_t, which provides specific information about the benefit of taking this particular action. This third channel differentiates our approach from traditional BC, which aims to predict the expert action directly from the state information, which would correspond to the first two channels in our setting. The target corresponding to this image is the resulting AUC estimate (8). From the perspective of reinforcement learning, this network can be viewed as a Q-network [17]; however, unlike Q-learning approaches, we are able to learn directly from an expert sampling policy, saving significant computation time and algorithm tuning.

IV. Empirical Results

To imitate the expert policy, we use a MobileNetV3 architecture [18], learning from 7,680 unique fields of size 30×30 generated by thresholding a two-dimensional GP with radial basis function kernel and a lengthscale of 0.2 over the domain $\mathcal{D} = [-1, 1] \times [-1, 1]$. We allow the sampler to travel a distance of $d_{\max} = 20$ units per field, resulting in approximately one million training examples. We compare our approach to sampling the location of maximum variance (VAR) and that of [2] (LUSR), choosing sample locations only among the 100 nearest neighbors in U_t at each round. We also compared to margin-based sampling and the approaches of [4] and [5], but we did not find any of these methods to consistently outperform VAR or LUSR and hence these results are omitted. The expert policy uses VAR as the base policy when computing the lookahead steps in (8).

Fig. 2 shows four example GP fields, where plots (a) and (b) (top row) correspond to fields with a lengthscale of 0.2, i.e., fields from the same distribution as the training data. These fields tend to have numerous connected components, some of which are only a few pixels in area, making LSE more challenging. The fields in plots (c) and (d) (bottom row) are drawn from a GP with lengthscale 0.5, resulting in fewer connected components and smoother boundaries. We first consider performance on 100 synthetic GP fields

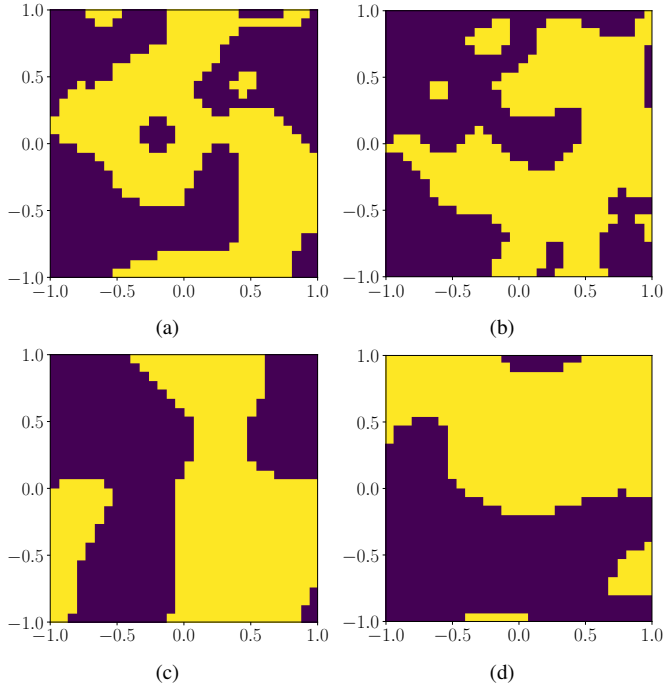


Fig. 2. Example level sets used for training and evaluation. Plots (a) and (b) are drawn from the training distribution, which is a GP with RBF kernel and lengthscale 0.2. Plots (c) and (d) are drawn from a GP with lengthscale 0.5.

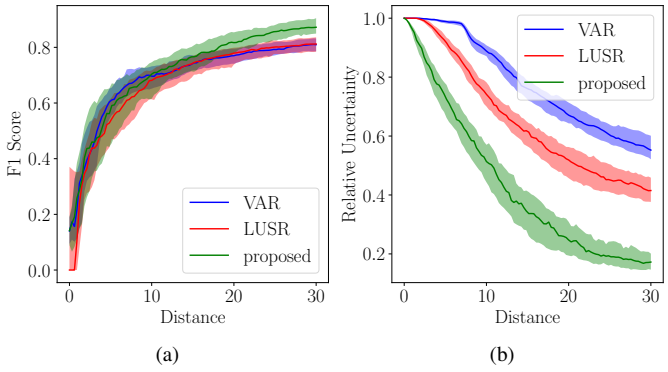


Fig. 3. Performance on synthetic GP fields with a lengthscale of 0.2 (same as training data). (a) F1 score as a function of distance. (b) Relative uncertainty as a function of distance.

with lengthscale 0.2. We generate similarities according to a truncated normal distribution with support $[0, 1]$, with means one and zero for within-class and across-class similarities, respectively, and a standard deviation of 0.05. We set $\eta = 0.5$ and the level set threshold $\tau = 0.5$. To ensure that we are testing scenarios not encountered by the expert policy, we allow the samplers to travel a distance of $d_{max} = 30$, i.e., 50% farther than the training data. Fig. 3(a) shows the median F1 score versus distance traveled along with the interquartile range. Our learned approach achieves the highest final F1 score, though the value is slightly lower in the initial stages. This behavior is due to the fact that the proposed approach "tracks" the boundary closely, as shown by an example sampling path in Fig. 6(b). The resulting AUC scores are 0.70 for VAR, 0.69 for LUSR, and 0.72 for our proposed approach, indicating

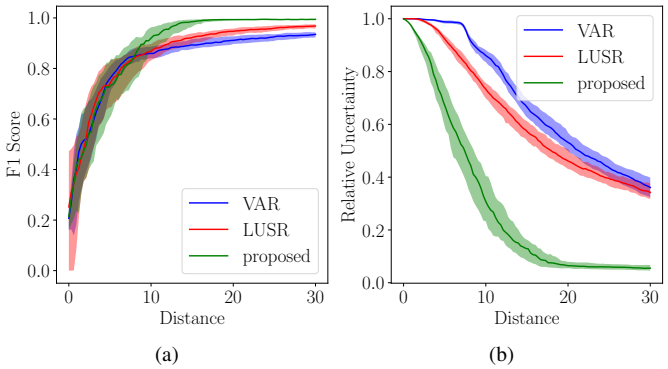
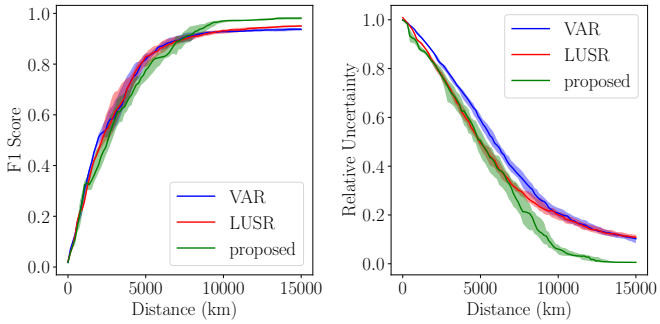


Fig. 4. Performance on synthetic GP fields with a lengthscale of 0.5 (different from training data). (a) F1 score as a function of distance. (b) Relative uncertainty as a function of distance.

only a minor performance benefit in terms of AUC. Fig. 3(b) shows the relative uncertainty versus distance and illustrates that our approach reduces uncertainty much more quickly than competing approaches. This is an important feature, as relative uncertainty may be used as a stopping criterion and is the only measurable proxy for estimation accuracy in real-world settings.

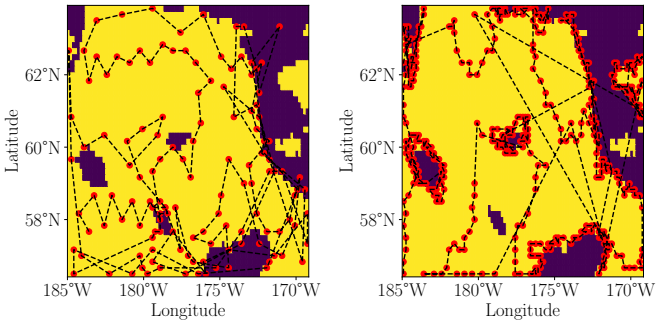
We next evaluate performance on synthetic fields with a lengthscale of 0.5, which is smoother than that seen in the training data. Fig. 4(a) shows the median F1 score versus distance traveled for all three methods. On these simpler level sets, our learned approach yields a significant performance improvement, obtaining an F1 score above 0.99 after traveling 16.36 units. In contrast, neither VAR nor LUSR obtains this score even after traveling the full maximum distance. The resulting AUCs are 0.83, 0.86, and 0.88 for VAR, LUSR, and our learned approach, respectively. This figure demonstrates the important fact that our approach provides strong performance even on states that are not seen in the training data, making BC a practical approach for this problem. Fig. 4 shows the relative uncertainty versus distance traveled and indicates that our approach reduces uncertainty very rapidly on these simpler level sets, dramatically outperforming competing approaches.

Finally, we demonstrate the performance of our approach on realistic ambient acoustic data generated using the multidimensional ambient noise model (MDANM) [19], which uses the Harrison model [20] to simulate ambient sound for ocean environments. We use an array of size $M = 32$, set the signal-to-noise ratio to 10, and use $L = 1500$ snapshots per location when calculating the sample covariance matrices. We consider a region in the northern Pacific Ocean containing HFEVA sediment types 3, 9, 17, 18, and 23. Using the bottom left corner (type 23, clay) as a reference location, we set the level set threshold $\tau = 0.5$ and the similarity parameter $\ell = 3$, so that the sublevel set consists of types $\{17, 18, 23\}$. As shown in [21], these sediment types are nearly indistinguishable from an information-theoretic standpoint. The region of interest and resulting sublevel/superlevel sets are shown in Fig. 1. We set $\eta = 0.08$ and evaluate performance over 32 random instances of collected samples. Fig. 5 shows (a) the F1 score and (b) relative



(a) (b)

Fig. 5. Performance on seabed data from the northern Pacific Ocean. (a) F1 score as a function of distance. (b) Relative uncertainty as a function of distance.



(a) (b)

Fig. 6. Example adaptive sampling paths on real-world sediment type data from the northern Pacific Ocean. Sample locations (red dots) and path (dashed line) are shown for (a) state-of-the-art approach LUSR [2] and (b) proposed learning-based approach.

uncertainty as a function of distance traveled on this dataset. Our method again obtains higher accuracy and significantly lower uncertainty for the same distance traveled, despite the fact that these similarities do not follow a truncated normal distribution and there is no guarantee the level set boundaries are well represented in the training data. Hence, we conclude that our approach successfully utilizes the GP model to train a robust BC agent. Example sampling paths for (a) LUSR and (b) our approach are shown in Fig. 6. While LUSR does focus samples near the level set boundaries. This "tracking" behavior is likely due to the inclusion of lookahead steps in our expert policy, which allows for nonmyopic treatment of the distance traveled.

V. Conclusion

We have demonstrated that a form of reinforcement learning known as behavioral cloning can be used to train an agent to efficiently classify large regions of the ocean according to seabed type. The success of this process relies on (1) developing an expert policy to imitate, and (2) including action-specific information as features when training our model. Our proposed method outperforms the state-of-the-art on both synthetic datasets and realistic ambient acoustic data, even when evaluated on data that differs significantly from the training set. To accomplish this, we transform seabed classification into a level set estimation problem, which is essentially binary classification. Extending our approach to handle multiclass classification settings is an important topic for future research.

References

[1] N. R. Chapman and E. C. Shang, "Review of geoacoustic inversion in underwater acoustics," *Journal of Theoretical and Computational Acoustics*, vol. 29, no. 03, p. 2130004, 2021.

[2] M. Sullivan, J. Gebbie, and J. Lipor, "Adaptive sampling for seabed identification from ambient acoustic noise," in *2023 IEEE 9th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*. IEEE, 2023, pp. 91–95.

[3] Naval Oceanographic Office Acoustics Division, "Database description for bottom sediment type (U)," 2003.

[4] A. Gotovos, N. Casati, G. Hitz, and A. Krause, "Active learning for level set estimation," in *IJCAI*, 2013, pp. 1344–1350.

[5] I. Bogunovic, J. Scarlett, A. Krause, and V. Cevher, "Truncated variance reduction: A unified approach to bayesian optimization and level-set estimation," in *Advances in neural information processing systems*, 2016, pp. 1507–1515.

[6] D. LeJeune, G. Dasarathy, and R. Baraniuk, "Thresholding graph bandits with grapl," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2476–2485.

[7] Q. P. Nguyen, B. K. H. Low, and P. Jaillet, "An information-theoretic framework for unifying active learning problems," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 10, 2021, pp. 9126–9134.

[8] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2, no. 3.

[9] F. Zhang, *The Schur complement and its applications*. Springer Science & Business Media, 2006, vol. 4.

[10] M. Valko, N. Korda, R. Munos, I. Flaounas, and N. Cristianini, "Finite-time analysis of kernelised contextual bandits," in *Uncertainty in Artificial Intelligence*, 2013.

[11] B. Bryan, R. C. Nichol, C. R. Genovese, J. Schneider, C. J. Miller, and L. Wasserman, "Active learning for identifying function threshold boundaries," *Advances in neural information processing systems*, vol. 18, 2005.

[12] P. Kearns, B. Jedynak, and J. Lipor, "A finite-horizon approach to active level set estimation," *Foundations of Data Science*, pp. 0–0, 2024.

[13] D. Wang, G. Dasarathy, and J. Lipor, "Distance-penalized active learning via markov decision processes," in *Proc. IEEE Data Science Workshop*, 2019.

[14] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne, "Imitation learning: A survey of learning methods," *ACM Computing Surveys (CSUR)*, vol. 50, no. 2, pp. 1–35, 2017.

[15] M. Zare, P. M. Kebria, A. Khosravi, and S. Nahavandi, "A survey of imitation learning: Algorithms, recent developments, and challenges," *IEEE Transactions on Cybernetics*, pp. 7173–7186, 2024.

[16] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena scientific Belmont, MA, 2005, vol. 1.

[17] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[18] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan *et al.*, "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 1314–1324.

[19] Naval Oceanographic Office Acoustics Division, "Software design description and software test description for the multi-dimensional ambient noise model (MDANM) version 1.01 (U)," 2022.

[20] C. Harrison, "Formulas for ambient noise level and coherence," *The journal of the acoustical society of America*, vol. 99, no. 4, pp. 2055–2066, 1996.

[21] J. Lipor, J. Gebbie, and M. Siderius, "On the limits of distinguishing seabed types via ambient acoustic sound," *The Journal of the Acoustical Society of America*, vol. 154, no. 5, pp. 2892–2903, 2023.