In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2022)*, Philadelphia, PA, USA May 2020

VOILA: Visual-Observation-Only Imitation Learning for Autonomous Navigation

Haresh Karnan[1], Garrett Warnell[2,3], Xuesu Xiao[3] and Peter Stone[3,4]

***Abstract*— While imitation learning for vision-based autonomous mobile robot navigation has recently received a great deal of attention in the research community, existing approaches typically require state-action demonstrations that were gathered using the deployment platform. However, what if one cannot easily outfit their platform to record these demonstration signals or—worse yet—the demonstrator does not have access to the platform at all? Is imitation learning for vision-based autonomous navigation even possible in such scenarios? In this work, we hypothesize that the answer is yes and that recent ideas from the Imitation from Observation (IfO) literature can be brought to bear such that a robot can learn to navigate using only ego-centric video collected by a demonstrator, even in the presence of viewpoint mismatch. To this end, we introduce a new algorithm, Visual-Observation-only Imitation Learning for Autonomous navigation (VOILA), that can successfully learn navigation policies from a single video demonstration collected from a physically different agent. We evaluate VOILA in the AirSim simulator and show that VOILA not only successfully imitates the expert, but that it also learns navigation policies that can generalize to novel environments. Further, we demonstrate the effectiveness of VOILA in a real-world setting by showing that it allows a wheeled Jackal robot to successfully imitate a human walking in an environment while recording video with a handheld mobile phone camera.**

I. INTRODUCTION

Enabling vision-based autonomous robot navigation has recently been a topic of great interest in the robotics and machine learning community [1]–[3]. Imitation learning in particular has emerged as a useful paradigm for designing vision-based navigation controllers. Using this paradigm, the desired navigation behavior is first demonstrated by another agent (usually a human), and then a recording of that behavior is supplied as training data to a machine learner that tries to find a control policy that can mimic the demonstration. To date, most approaches in the navigation domain that use imitation learning require demonstration recordings that contain both state observations (e.g., images) and actions (e.g., steering wheel angle or acceleration) gathered onboard the deployment platform [1], [2], [4], [5].

*This work has taken place in the Learning Agents Research Group (LARG) at UT Austin. LARG research is supported in part by NSF (CPS-1739964, IIS-1724157, NRI-1925082), ONR (N00014-18-2243), FLI (RFP2-000), ARO (W911NF-19-2-0333), DARPA, Lockheed Martin, GM, and Bosch. Peter Stone serves as the Executive Director of Sony AI America and receives financial compensation for this work. The terms of this arrangement have been reviewed and approved by the University of Texas at Austin in accordance with its policy on objectivity in research.

[1] The University of Texas at Austin, Department of Mechanical Engineering haresh.miriyala@utexas.edu

[2] Army Research Laboratory garrett.a.warnell.civ@mail.mil

[3] The University of Texas at Austin, Department of Computer Science xiao@cs.utexas.edu

[4] Sony AI pstone@cs.utexas.edu

While these existing imitation learning approaches have proved successful in certain scenarios, there are situations in which it would be beneficial to relax the requirements they impose on the demonstration data. For example, if we wish to collect a large number of demonstrations from many experts, it may prove too difficult or costly to arrange for each expert to operate specific deployment platforms, which are often expensive or difficult to transport. Additionally, it might be costly to outfit all demonstration platforms with instrumentation to record the control signals with the demonstration data. However, due to the low cost and portability of video cameras, it may still be feasible to have demonstrators record ego-centric video demonstrations of their navigation behaviors while operating a different platform. Demonstrations of this nature would consist of video observations only (i.e., they would not contain control signals), and, because of the difference in platform, the videos would likely exhibit ego-centric viewpoint mismatch compared to those that would be captured by the deployment platform. One example of such data is the plethora of vehicle dashcam videos available in publicly-accessible databases [6] or on YouTube. Another example is video demonstrations of robot behaviors generated by proprietary code that one would like to mimic on the same or different robot hardware. Unfortunately, to the best of our knowledge, there exist no current imitation learning techniques for vision-based navigation that can leverage such demonstration data.

Fortunately, recent work in *Imitation from Observation* (IfO) [7]—imitation learning in the absence of demonstrator actions—has shown a great deal of success for several related tasks. For example, work in this area has been able to learn from video-only demonstrations for both simulated and real limbed robots [8]–[11]. However, no literature of which we are aware has considered whether these IfO techniques can be applied to the vision-based autonomous navigation problem we have outlined above. This problem is especially challenging since physical differences in the demonstration platform introduce viewpoint mismatch in the video demonstrations.

In this paper, we hypothesize that it is possible to perform imitation learning for vision-based autonomous navigation using video-only demonstrations collected using a physically different platform. To this end, we introduce a new IfO technique for vision-based autonomous navigation called *Visual-Observation-only Imitation Learning for Autonomous navigation* (VOILA).[1] To overcome viewpoint mismatch, VOILA uses a novel reward function that relies on off-

[1] A preliminary version of this work was presented at the 2021 AAAI Spring Symposium on Machine Learning for Navigation.

Fig. 1: Policy rollout trajectories of the VOILA agent (green) successfully imitating a demonstration behavior (black) of patrolling a rectangular hallway clockwise. The demonstration consists of a video gathered by a human walking while using a handheld camera that is considerably higher than the robot's camera (introducing significant viewpoint mismatch). We see that the VOILA agent is able to successfully imitate the expert demonstration even in the presence of this egocentric viewpoint mismatch.

the-shelf keypoint detection algorithms that are themselves designed to be robust to egocentric viewpoint mismatch. This novel reward function is utilized to drive a reinforcement learning procedure that results in navigation policies that imitate the demonstrator.

We experimentally confirm our hypothesis both in simulation and on a physical Clearpath Jackal robot. We compare VOILA against a state-of-the-art IfO algorithm GAIfO [9], and show that VOILA can learn to imitate an expert's visual demonstration in the presence of viewpoint mismatch while also generalizing to environments not seen during training. Additionally, we demonstrate the flexibility of VOILA by showing that it can also support vision-based training of navigation policies with inputs other than camera images.

II. Background and Related Work

The proposed approach, VOILA, performs reinforcement learning (RL) using a novel reward function based on image keypoints in order to accomplish imitation from observation for autonomous navigation with viewpoint mismatch. In this section, we review related work in autonomous robot navigation, imitation from observation, and in computer vision techniques for visual feature extraction.

A. Machine Learning for Autonomous Navigation

The use of machine learning methods in the design of autonomous navigation systems goes back several decades, though recent years have seen a spike in interest from the research community [2], [3], [12]–[14]. One of the earliest successes was reported by Pomerleau [15], in which a system called ALVINN used imitation learning to train an artificial neural network that could perform lane keeping based on demonstration data generated in simulation. Since then, several improvements, both in the amount and type of demonstration data and in network architecture and training, have been proposed in the literature. In particular, LeCun *et al.* proposed the use of a convolutional neural network (CNN) to better process real demonstration images for an off-road driving task [4], and, more recently, Bojarski *et al.* reported that gathering a large amount of real-world human driving demonstration data and applying data augmentation made it possible to train even more-complex CNN architectures to perform lane keeping [1].

While the aforementioned approaches each use end-to-end imitation learning to find autonomous navigation policies, other machine learning for autonomous navigation work has adopted the alternative training paradigm of RL. Chang *et al.* propose using off-policy Q-learning from video demonstration data to learn goal conditioned hierarchical policies for semantic navigation [14]. While their approach also uses video demonstrations with no action labels, they learn a goal conditioned policy with access to thousands of navigation video examples whereas in our work, the focus is on imitating an expert's video demonstration in the presence of viewpoint mismatch using a single video-only demonstration. Gupta *et al.* [16] propose a context translation network to imitate an expert demonstration in the presence of viewpoint mismatch. However, their approach requires multiple demonstrations with differing camera viewpoints in each demonstration to provide context signals. Additionally, their approach only deals with third-person viewpoint mismatch and does not consider the egocentric viewpoint mismatch problem that is considered in this work. Similarly, Gaskett *et al.* [17] propose a visual servoing algorithm that uses image template-matching to provide rewards for their reactive agent. However, their algorithm is tailored towards visual servoing, whereas this work focuses on robot

navigation.

The work closest to ours is that of Kendall *et al.* [18], in which the proposed system learns a navigation policy using RL, where the reward function is the total distance travelled by their autonomous vehicle before a human driver intervenes (to, e.g., prevent collisions). However, unlike VOILA, Kendall *et al.* utilize experience gathered exclusively by the learning platform itself, considering neither imitation from observation nor the particular problem of viewpoint mismatch.

B. Imitation from Observation

Recently, there have been a number of imitation from observation (IfO) techniques introduced in the literature, including the adversarial approach, GAIfO, proposed by Torabi *et al.* [9], [19] which we use as a comparison point in this paper. In GAIfO, the reward signal is provided by a learned discriminator network which seeks to reward state transitions similar to those present in the demonstration and penalize—if it can tell the difference—state transitions that come from the imitator. While GAIFO has been shown to be successful in both low- and high-dimensional observation spaces, it has thus far only been applied to continuous control tasks for limbed agents. Moreover, as we will show in our experiments, while GAIfO is able to imitate the expert when the egocentric viewpoints between the expert and the imitator match, it is unable to do so in the presence of viewpoint mismatch.

Viewpoint mismatch in IfO has been previously considered in the work by Sermanet *et al.* [10], which proposes Time Contrastive Networks (TCNs). TCNs use a triplet loss metric to learn a feature space embedding which is then used for rewarding the agent to imitate the expert. While both VOILA and TCNs are robust to viewpoint mismatch, TCNs require demonstration data with multiple viewpoints in the same timestep in order to learn an embedding space robust to viewpoint mismatch, whereas VOILA achieves this robustness by leveraging feature detection algorithms (e.g SIFT [20]) commonly used in SLAM that are themselves designed to be robust to viewpoint mismatch.

C. Feature Detection and Matching

To overcome viewpoint mismatch, VOILA utilizes a novel reward function that relies on local image features such as keypoints and their descriptors. Keypoints have been used for decades to solve challenging tasks such as image verification, matching and retrieval. More recently, deep-learning-based keypoint extractors such as SUPERPOINT [21] have been shown to be more successful than classical approaches. In this work, we use SUPERPOINT to detect keypoints and their corresponding descriptors, and we determine keypoint matches between two images using the typical method based on the two nearest neighbors in descriptor space [21] using the ℓ_2 distance metric. However, in principle VOILA can be used with any local feature detector or feature matching algorithm. In our experiments, we use the simplest image retrieval algorithm of local keypoint detection followed by nearest neighbor matching in the descriptor space to identify the visually closest image with maximum keypoint matches. However, there exist more sophisticated algorithms such as NetVLAD [22], [23] that are more robust to visual aliasing errors for the task of image retrieval which can be used with VOILA with no changes to the underlying algorithm.

Several works have proposed learning a keypoint detector specific to the imitation learning task [24], [25]. Unlike such approaches, VOILA uses an off-the-shelf keypoint extractor that is not trained specifically for the navigation task.

D. SLAM-based Approaches for Navigation

Visual Teach and Repeat (VTR) methods such as the one proposed by Barfoot *et al.* [26] follow a two step approach to imitating a navigation demonstration. First, a SLAM map of the demonstration environment is built in the *teach* phase. In the subsequent *repeat* phase, the robot localizes within a submap of the environment and follows a desired trajectory. VOILA, on the other hand, learns a reactive navigation policy directly from demonstrations, sidestepping the SLAM problem.

III. VOILA

In this section, we formulate the imitation learning problem for the task of autonomous visual navigation, which we pose as a reinforcement learning problem with a demonstration-dependent reward. The critical contribution of VOILA is the development of this particular reward function, which we describe in detail below.

A. Preliminaries

We treat autonomous visual navigation as a RL problem where the environment is a Markov decision process. At every time step t, the state of the agent is described by $s_t \in \mathcal{S}$, the observation of the agent is described by $O_t \in \mathcal{O}$, and an action $a_t \in \mathcal{A}$ is sampled from the agent's policy $a_t \sim \pi(\cdot|O_t)$.[2] A single expert demonstration is represented as a set of n sequential observations $\mathcal{D}^e = \{I_1, I_2, \ldots, I_n\}$. Performing this action in the environment leads to a next state $s_{t+1} \sim T(\cdot|s_t, a_t)$, where T is the unknown transition dynamics of the agent in the environment. For this specific transition, the agent receives a reward, $r_{t+1} \in \mathbb{R}$, which is a function of both the agent's transition tuple and the demonstration, i.e., $r_{t+1} = R(O_t, a_t, O_{t+1}; \mathcal{D}^e)$. The relative utility of near-term and long-term reward is controlled using the discount factor $\gamma \in (0, 1]$. The RL objective is to find a policy π that maximizes the expected sum of discounted rewards $\mathbb{E}[\Sigma_{t=0}^{\infty} \gamma^t R(O_t, a_t, O_{t+1}; \mathcal{D}^e)]$.

B. Reward Formulation for Imitation Learning

For each transition (O_t, a_t, O_{t+1}) experienced by the learner, we require a reward r_{t+1} such that the learner, by optimizing the RL objective with this reward, can learn to

[2]While VOILA's reward function depends on camera images, the imitation policy can actually be learned over *any* appropriate state representation—vision-based or otherwise. We show one such example using LiDAR scans as the state representation in Fig. 1.

imitate the demonstration. In particular, because we wish to perform learning in real time, we seek a dense reward function that provides feedback at each timestep without delay. Since the expert demonstrations are from a physically different agent, there can be significant ego-centric viewpoint mismatch between the observation spaces of the learner and the demonstrator as shown in Fig. 1. Such a mismatch poses a challenge to designing a good reward function since it is not immediately clear how to compare images from different viewpoints. Hence, we introduce here a novel reward function based on keypoint feature matches between the expert and the imitator's ego-centric observations for the task of imitation learning for visual navigation. Keypoint detectors have been extensively used in the computer vision community for several decades to solve challenging tasks like structure-from-motion (SfM), visual SLAM and hierarchical localization. Recent keypoint detection algorithms like SUPERPOINT [21] provide invariance to perspective distortion, scaling, translation, rotation, viewpoint mismatches, and varied lighting conditions between the key and query images. Hence, we use keypoint detectors to help define the reward function to learn visual navigation policies from demonstrations provided by any other agent.

The reward function we propose relies on a quantity that we call *match density*. We define the match density $d(O_1, O_2)$ between two images O_1 and O_2 as the ratio of the number of keypoint matches between O_1 and O_2, and the total number of detected keypoints in O_2. $d(O_1, O_2) \in [0, 1]$, assuming there is always a non-zero number of keypoints detected in an image. Additionally, instead of imposing a temporal alignment constraint, we define the reward for a particular transition by searching the demonstration for the image which is most visually similar to the learner's current observation. Here, we define the most visually similar image in the expert demonstration to be the one that has the highest match density with O_t, which we denote as I_t. Fig. 1 shows the imitator's current observation O_t and corresponding visually closest expert image I_t. For convenience of notation, we denote the next image after I_t in the demonstration sequence as J_t.

Using the concepts described above, we now define the proposed reward function for VOILA:

$$R(O_t, a_t, O_{t+1}; \mathcal{D}^e) = \begin{cases} F + V - \lambda ||a_t^{steer}|| & , alive \\ -10 & , done \end{cases}, \tag{1}$$

where $F = d(O_{t+1}, I_{t+1})$ and $V = \gamma * d(O_{t+1}, J_t) - d(O_t, J_t)$. If the robot is in the *done* state, i.e., it has crashed (as detected in AirSim, or by the trainer in physical experiments) or the number of keypoint matches drop below 10, the agent receives a penalty reward of -10. Otherwise, the agent is in the $alive$ state, and we assign a reward that depends on terms F and V. The F term assigns reward value based on the match density encountered at the next observation O_{t+1} that the agent ends up in the transition. This component encourages the agent to stay on the demonstrated trajectory. The V term is similar to a potential-based shaping term, and rewards a transition based on the difference in the match densities with the next expert observation J_t and the imitator's observations. This component encourages the imitator to find a policy that exhibits similar state transitions to those experienced by the expert. We additionally found that adding the action penalty term with a λ of 0.01 penalizes the agent for making large steering changes. Since the maximum value of the reward function in the *alive* state is 2, we heuristically picked a value 5 times higher and arrived at -10 for the negative penalty. This value worked well empirically in our experiments; we do not have any reason to believe that the results are particularly sensitive to this exact value. The expert image retrieval step is performed in real-time using feature matching, and is outlined in the implementation section.

IV. Implementation

In this section, we provide specific implementation details of VOILA including those related to representation learning, keypoint feature extraction, and the network architectures.

A. Representation Learning

Representation learning using unsupervised learning is a powerful tool to improve the sample efficiency of deep RL algorithms. Instead of learning a navigation policy over high dimensional image space, VOILA uses a latent representation of the image and learns the navigation policy over this latent code as input to the policy. Specifically, VOILA uses a Regularized Auto Encoder (RAE) [27] to learn a latent posterior of the visual observations of the imitator. The imitating control policy is then learned using RL with the latent code $z_t = g_\phi(O_t)$ as the input to the policy network, where g_ϕ is the encoder of the RAE with weights ϕ. A ResNet-18 encoder-decoder network architecture is used for the RAE and is trained for the task of image reconstruction, with data collected from the imitator using random rollouts. The input images are of size 256×256 and the size of the latent dimension is 512. Random cropping and random affine image augmentations are utilized to regularize training.

B. Keypoint Feature Extraction

As a preprocessing step, all SUPERPOINT features detected from expert observations are stored in a buffer. At the start of an episode (for the first frame), the nearest expert observation I_t to O_t is retrieved by linearly searching for the closest expert observation in $\mathcal{D}^e$ with the maximum feature matches. As the episode unfolds, instead of exhaustively searching for the closest expert image at every transition, the search is restricted over the three next expert observations forward in time from the previous closest expert image in $\mathcal{D}^e$. At each transition, SUPERPOINT keypoints and descriptors are extracted and used to retrieve the closest expert image and compute the reward according to Equation 1. Note that the SUPERPOINT keypoint descriptors extracted in an image are the local features and not the global features for that image. Hence, we explicitly train an RAE to learn a

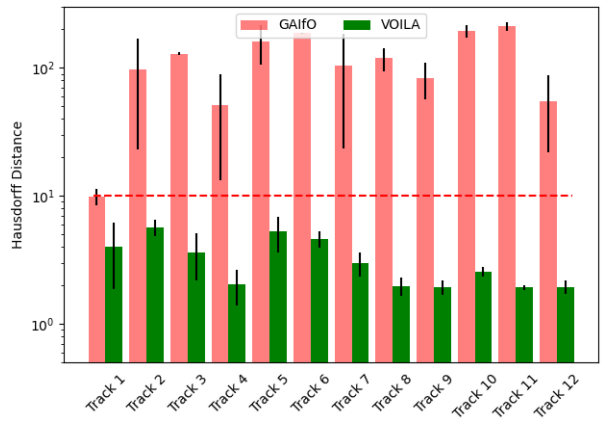

Fig. 2: Imitation performance of policies learned using VOILA and GAIfO in AirSim. The y-axis shows Hausdorff distance between expert and imitator's trajectories, averaged across five trials (lower distance indicates behavior more similar to the expert). A Hausdorff distance greater than 10.0 (marked by the red line) indicates a failure in imitating the demonstration. We see that with viewpoint mismatch, the GAIfO agent is unable to imitate the expert successfully on all tracks, whereas VOILA is unaffected by viewpoint mismatch and results in policies that induce behavior closer to that of the demonstrator.

compressed global representation of the image for training the navigation policy, as described in the previous section. The SUPERPOINT descriptor [21] for every keypoint is a vector of 512 float values. These descriptors are learned from data and correspond to different features of a keypoint including—but not limited to—lighting, location, appearance and texture.

C. Navigation Policy Architecture

We model the navigation policy π using a 3-layer, fully-connected neural network, with 256 neurons per layer. We perform frame-stacking with two consecutive latent codes of observations in time to alleviate effects of partial observability in the environment. We also additionally include the most recent action performed by the agent as a part of the state. We use Soft Actor-Critic (SAC) [28], an off-policy RL algorithm, to learn π.

V. EXPERIMENTS

We now describe the experiments that we performed to evaluate VOILA. The experiments are designed to answer the following questions:

(Q_1) Is VOILA capable of learning imitative policies from video demonstrations that exhibit viewpoint mismatch?

(Q_2) How well do policies learned using VOILA generalize to environments unseen during training?

To answer the questions above, we perform experiments using both a simulated autonomous vehicle and a real Clearpath Jackal robot on the task of imitating "road following" and "hallway patrol" tasks, respectively. The objective of the VOILA agent is to imitate the expert's visual demonstration by learning an end-to-end navigation policy, even in the presence of viewpoint mismatch. To quantify performance of imitation policies, we compute the Hausdorff distance metric (lower is better) between trajectories generated on a held-out set of environments.

A. Simulation Experiments in AirSim

In our simulation experiments, we answer questions Q_1 and Q_2 using the outdoor 'Neighborhood' environment in AirSim [29] and learn the task of road following, i.e., driving on a straight road avoiding collisions with obstacles such as parked cars along the curb while dealing with varied lightning conditions along its path. To this end, we pick 12 straight road segments (tracks) in the AirSim environment. Two tracks (Track 1 and 2) were used for training the agent, and the learned policy is deployed on all 12 tracks. The other 10 tracks and their expert demonstrations are not seen by the agent prior to evaluation, and so we use them to test the generalizability of the learned policy to unseen environments. In each episode, the car begins at a randomized initial position near the start of the track, so the agent cannot trivially solve the task by learning to drive straight without having to steer. The expert demonstrations consist of a single trajectory (egocentric, front-facing images) for all tracks provided by a human (the first author) controlling the demonstration vehicle with the objective of navigating from start to end of the tracks, driving straight, in the middle of the road, and avoiding collisions with obstacles such as parked cars.

We use the latent vector of the RAE as the state representation, and the action space consists of change in steering and throttle values. Note that the expert demonstrations are required only during training to compute the reward. At test time, the agent imitates the expert without requiring access to expert demonstrations.

We compare VOILA against GAIfO, a state-of-the-art IfO algorithm that does not explicitly seek to overcome viewpoint mismatch. While both GAIfO and VOILA are IfO algorithms that can imitate from video-only demonstration data, GAIfO has been evaluated predominantly in domains such as limbed-robot locomotion and manipulation, whereas VOILA has been designed specifically for vehicle navigation domains. Additionally, GAIfO uses a learned reward function whereas in VOILA, we propose a manually defined reward function that is not learned. To ensure a fair comparison, we provide each algorithm the same state representation, i.e., the latent code of the RAE. Further, since GAIfO is an on-policy algorithm whereas VOILA relies on the off-policy SAC algorithm, we allow GAIfO ten times more training timesteps than VOILA (1 million vs. 100,000). Finally, we report results for GAIfO using the policy that achieved maximum on-policy returns during training.

Fig. 2 addresses both Q_1 and Q_2. In Fig. 2, in the presence of viewpoint mismatch, we see that, as expected, GAIfO is unable to imitate the expert on the training Track 2 and does not generalize to other environments. However, confirming our hypothesis, VOILA is able to imitate the

expert demonstration even in the presence of viewpoint mismatch on seen Tracks 1 and 2 and also generalizes to the other 10 unseen tracks.

B. Physical Experiments on the Jackal

To answer Q_1 and Q_2 on a physical robot, we performed experiments using a Clearpath Jackal—a four-wheeled, differential drive ground robot equipped with a front facing camera. The environment considered is an indoor office space, shown in Fig. 1, consisting of carpeted floors, straight hallways, intersections, and turns. There are also static obstacles such as benches, chairs, whiteboards, pillars along the wall, and trashcans, all of which the robot needs to avoid colliding with.

We evaluated VOILA on a hallway patrol task, in which the robot begins at a start state (shown in Fig. 1) and patrols around the building clockwise by taking the first right at intersections and driving straight in the hallways. To obtain a video demonstration of this task from a physically different agent, a human (the first author) walked the patrol trajectory once while recording video using a mobile phone camera held approximately 4 feet above the ground (the robot's camera is at approximately 0.8 feet from the ground). To contrast the imitation learning performance of VOILA with and without any viewpoint mismatch, we perform additional experiments (see full version of paper on ArXiv [30]), henceforth called VOILA-w/o-mismatch in which the expert demonstrations are collected onboard the deployment platform itself. These demonstrations are collected using the ROS `move_base` [31] navigation stack with a pre-built map of the environment and waypoints to patrol the environment while recording the egocentric visual observations from the front facing camera.

A VOILA training episode consists of the robot starting at approximately the same start state (as shown in Fig. 1) and exploring in the training environment until the agent reaches the *done* state. After each training episode, the robot was manually reset back to the start state by a human operator, and a new training episode began. Training the navigation policy happened onboard the robot on a GTX 1050Ti GPU.

The VOILA agent trained using demonstrations with and without viewpoint mismatch learned to imitate the expert within 90 minutes (120 episodes) and 60 minutes (100 episodes), respectively, of experiment time (including time taken to reset the robot at the end of an episode). Fig. 1 shows the trajectory rollout (in green) of the policy learned using VOILA, imitating the expert demonstration in the presence of viewpoint mismatch. Addressing Q_1, we see that VOILA is able to successfully patrol the indoor environment in a real-world setting, as demonstrated by a physically different expert agent, in the presence of viewpoint mismatch.

To evaluate the generalizability of policies learned using VOILA to unseen real-world conditions (Q_2), we deploy the policy learned by VOILA in a 'perturbed environment', in which positions of movable objects such as trashcans, doors, whiteboards, chairs, and benches in the training environment are perturbed as shown in Fig. 3. We see that, even with such

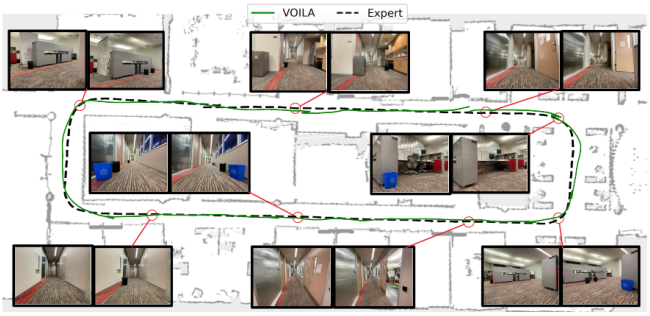

Fig. 3: The VOILA agent (green), trained in the unperturbed training environment (left), deployed here in the perturbed environment (right). We see that the learned policy is robust to the visual differences between the training and deployment environment, examples of which are provided as image pairs.

environmental changes, VOILA is able to successfully patrol the hallway without any collisions. To quantify the imitation learning performance of VOILA, we compute the Hausdorff distance between the human demonstrated trajectory and the trajectory generated by the policy learned using VOILA. The Hausdorff distance between the human demonstration and VOILA is 0.783. The VOILA-w/o-mismatch agent trained using demonstrations without any egocentric viewpoint mismatch achieves a Hausdorff distance of 0.665. To provide context for the Hausdorff distance metric, we also compute the Hausdorff distance for suboptimal and random trajectories. The suboptimal trajectory was collected by navigating along the hallway in a zig-zag route using `move_base` by setting waypoints closer to the walls and achieves a Hausdorff distance of 0.806. The random trajectory was collected using a randomly initialized policy, which fails quickly by crashing in the environment, achieving a Hausdorff distance of 1.192. We see that the VOILA agent is able to successfully imitate the expert's video-only navigation demonstration and patrols the hallway without any collisions.

VI. Conclusion and Future Work

In this paper, we introduced Visual-Observation-only Imitation Learning for Autonomous navigation (VOILA), a new approach that enables imitation learning for autonomous robot navigation using a single, egocentric, video-only demonstration while being robust to egocentric viewpoint mismatch. VOILA formulates the imitation problem as one of reinforcement learning using a novel reward function that is based on keypoint matches between the expert and imitator's visual observations. We showed through experiments, both in simulation and on a physical robot, that, by optimizing the proposed reward function using reinforcement learning, VOILA could successfully find a good imitation policy that maps sensor observations directly to low level action commands. We additionally performed experiments that tested the generalizability of policies trained using VOILA to unseen environments. One interesting direction for future work is to explore state representations that enable VOILA to generalize better across environments.

REFERENCES

[1] M. Bojarski, D. D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba, "End to end learning for self-driving cars."

[2] F. Codevilla, M. Müller, A. Dosovitskiy, A. M. López, and V. Koltun, "End-to-end driving via conditional imitation learning," 2017.

[3] Y. Pan, C.-A. Cheng, K. Saigol, K. Lee, X. Yan, E. Theodorou, and B. Boots, "Agile autonomous driving using end-to-end deep imitation learning," in *Robotics: Science and Systems*, 2018.

[4] Y. Lecun, U. Muller, J. Ben, E. Cosatto, and B. Flepp, "Off-road obstacle avoidance through end-to-end learning." 01 2005.

[5] A. Tampuu, M. Semikin, N. Muhammad, D. Fishman, and T. Matiisen, "A survey of end-to-end driving: Architectures and training methods."

[6] F.-H. Chan, "Anticipating Accidents in Dashcam Videos," https://aliensunmin.github.io/project/dashcam/, 2017.

[7] F. Torabi, G. Warnell, and P. Stone, "Recent advances in imitation learning from observation," 2019.

[8] B. S. Pavse, F. Torabi, J. P. Hanna, G. Warnell, and P. Stone, "RIDM: reinforced inverse dynamics modeling for learning from a single observed demonstration," *CoRR*, vol. abs/1906.07372, 2019.

[9] F. Torabi, G. Warnell, and P. Stone, "Generative adversarial imitation from observation," *arXiv preprint arXiv:1807.06158*, 2018.

[10] P. Sermanet, C. Lynch, J. Hsu, and S. Levine, "Time-contrastive networks: Self-supervised learning from multi-view observation."

[11] X. Pan, T. Zhang, B. Ichter, A. Faust, J. Tan, and S. Ha, "Zero-shot imitation learning from demonstrations for legged robot visual navigation," *CoRR*, vol. abs/1909.12971, 2019. [Online]. Available: http://arxiv.org/abs/1909.12971

[12] J. Bi, T. Xiao, Q. Sun, and C. Xu, "Navigation by imitation in a pedestrian-rich environment," *ArXiv*, vol. abs/1811.00506, 2018.

[13] X. Xiao, B. Liu, G. Warnell, and P. Stone, "Motion control for mobile robot navigation using machine learning: a survey," 2020.

[14] M. Chang, A. Gupta, and S. Gupta, "Semantic visual navigation by watching youtube videos," *CoRR*, vol. abs/2006.10034, 2020. [Online]. Available: https://arxiv.org/abs/2006.10034

[15] D. A. Pomerleau, *ALVINN: An Autonomous Land Vehicle in a Neural Network*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1989, p. 305–313.

[16] Y. Liu, A. Gupta, P. Abbeel, and S. Levine, "Imitation from observation: Learning to imitate behaviors from raw video via context translation," *CoRR*, vol. abs/1707.03374, 2017. [Online]. Available: http://arxiv.org/abs/1707.03374

[17] C. Gaskett, L. Fletcher, and E. Zelinsky, "Reinforcement learning for visual servoing of a mobile robot," 09 2000.

[18] A. Kendall, J. Hawke, D. Janz, P. Mazur, D. Reda, J. Allen, V. Lam, A. Bewley, and A. Shah, "Learning to drive in a day," 2018.

[19] F. Torabi, G. Warnell, and P. Stone, "Imitation learning from video by leveraging proprioception," 2019.

[20] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, ser. ICCV '99. USA: IEEE Computer Society, 1999, p. 1150.

[21] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *CVPR Deep Learning for Visual SLAM Workshop*, 2018.

[22] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," 2016.

[23] P. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "From coarse to fine: Robust hierarchical localization at large scale," *CoRR*, vol. abs/1812.03506, 2018. [Online]. Available: http://arxiv.org/abs/1812.03506

[24] N. Das, S. Bechtle, T. Davchev, D. Jayaraman, A. Rai, and F. Meier, "Model-based inverse reinforcement learning from visual demonstrations," *CoRR*, vol. abs/2010.09034, 2020. [Online]. Available: https://arxiv.org/abs/2010.09034

[25] L. Manuelli, Y. Li, P. R. Florence, and R. Tedrake, "Keypoints into the future: Self-supervised correspondence in model-based reinforcement learning," *CoRR*, vol. abs/2009.05085, 2020. [Online]. Available: https://arxiv.org/abs/2009.05085

[26] P. Furgale and T. D. Barfoot, "Visual teach and repeat for long-range rover autonomy," *J. Field Robot.*, vol. 27, no. 5, p. 534–560, Sep. 2010.

[27] P. Ghosh, M. S. M. Sajjadi, A. Vergari, M. Black, and B. Scholkopf, "From variational to deterministic autoencoders," in *International Conference on Learning Representations*, 2020.

[28] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," 2018.

[29] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "Airsim: High-fidelity visual and physical simulation for autonomous vehicles," in *Field and Service Robotics*, 2017.

[30] H. Karnan, G. Warnell, X. Xiao, and P. Stone, "Voila: Visual-observation-only imitation learning for autonomous navigation," 2021.

[31] "ROS movebase navigation stack," http://wiki.ros.org/move_base, accessed: 2021-09-9.