Teaching a Robotic Guide Dog to Walk with Human User

Taery Kim   Yoonwoo Kim   John Seon Keun Yi

***Abstract*—Although guide dogs can have crucial influence on the lives of visually impaired people, training the dogs are resource heavy. Therefore, there have been several efforts to transition to robotic guide dog systems. In this paper, we teach a quadrupedal robot to walk with its human user by leveraging recent reinforcement learning techniques that teach quadrupedal robots to walk in the real world. Our method is divided into two separate steps namely, pertraining and finetuning. In the pretraining step, a locomotion policy is trained in simulation using a reference motion of a dog. In the finetuning step, we test two hypotheses. First, we test how stably a policy can walk when human gait force is applied. Second, we test if policy can be fine-tuned to follow the direction of the human gait force by introducing additional reward in the fine-tuning step. We observed that pre-training the motion imitation model with random perturbations is effective when the goal is to walk stably when human gait force is applied. However, when the goal changes to moving in the direction of the applied human gait force, due to the adversarial nature of the original reward and the newly introduced reward, the policy underperforms.**

I. Introduction

Guide dogs are widely used to navigate visually impaired people. These dogs take a lot of resources and time to train, and once deployed only can serve for a limited time [1]. There have been several efforts to resolve this problem by developing robotic guide systems [2]–[6]. However, these works use a leash or a rigid rod instead of a conventional rigid harness [2]–[4], or utilize a different robot form [3], [5], [6]. With the recent abundance of works that teach quadrupedal robots to walk in the real world [7]–[9], we focus on the most basic but integral task of walking. In this work, we use a quadrupedal robot with a rigid harness attached to its body, which is an identical setting to most guide dogs. Due to the rigidity of the harness, the dynamic force from the human arm holding the harness can have an effect on the robot gait as it is walking alongside the human. In this project, we develop a deep reinforcement learning model to teach the guide dog robot to walk naturally with external resistance. We adopt a previous work [10] that fine-tunes a locomotion policy in the real world and attempt to improve the learned policy by inserting external perturbations in the pre-training process in simulation. Furthermore, we attempt to expand the learned policy by fine-tuning the robot to walk while following feedback forces such as pushing and pulling. We achieve this by implementing new rewards that minimize the feedback force and introducing a perturbation classifier to distinguish human feedback from random perturbations. Our key contributions are as follows:

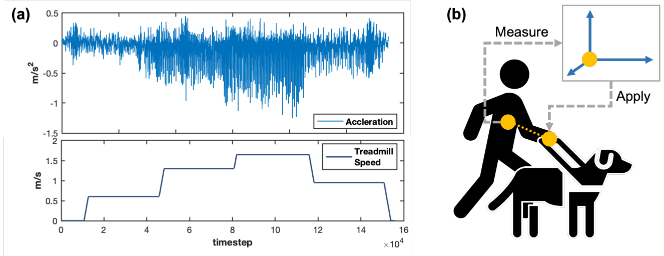

Fig. 1. Human gait modeling. (a) Example of the acceleration data. Upper plot shows the human's trunk acceleration in z-axis (forward direction) when treadmill speed changes as plotted in the bottom. (b) Diagram showing how the human's trunk acceleration delivered to the robot through harness.

- We formulate tasks for robot guide dogs that use rigid harness in RL, which is the first attempt, to the best of our knowledge.
- We propose new reward designs to additionally train policies to follow human feedback.
- We introduce a LSTM classifier to distinguish human feedback force.

II. Related Work

There have been ongoing efforts [7]–[9] to tackle locomotion tasks for quadrupedal robot using deep reinforcement learning. One major issue is learning a locomotion policy that can adapt to different environments sometimes unseen in training. Kumar et al. [11] attempts to achieve this by training the robot on different terrains in simluation, and training a separate adaptation module to predict the extrinsics using the history of observations and actions. Smith et al. [10] pre-trains basic locomotion tasks in simulation and fine-tunes it in the real world. We closely follow the structure of [10] but apply changes to our task of guide dog walking.

III. Method

Overall, we follow the work of Smith et al. [10] to pre-train a locomotion policy in simulation. We adapt this work to robotic guide dog locomotion by inserting external perturbations to the harness during pre-training, using the off-policy RL algorithm [12]. Instead of further fine-tuning the locomotion policy like the referenced paper, we aim to learn a new objective in the fine-tuning process. Our goal of fine-tuning is to follow the human feedback force while walking. Using the pre-trained locomotion policy as a base, we introduce two new rewards to encourage the robot to walk in the direction of the feedback force.

A. Human Gait Modeling

In this project, we use a 3-dimensional biomechanical and wearable sensor dataset [13] to simulate the force the robotic guide dog will receive when walking alongside a human. We will call this force the *human gait force (perturbation)*. Specifically, the acceleration of the human trunk when walking in various speeds (Fig. 1.a) is multiplied by the human's mass to create the force. If we assume human's upper body is a rigid body and the human is walking at the same speed with the robot, the force at human's trunk generated through walking can be applied directly to the harness and then to the robot (Fig. 1.b). The human gait force created from the data is used as one of the perturbations in pre-training, and also utilized to evaluate the learned policies.

B. Pre-training Locomotion

The goal of the pre-training process is similar to that of [10]: to learn agile robotic locomotion skills from reference motion in simulation. While the previous work tackles multiple tasks such as pacing backwards or taking side-steps, we focus on learning a policy for forward pacing.

The policy is trained by imitating the provided reference motion clips from [14] using a reward function that encourages tracking the target poses at each time step. The adopted reward function at each timestep t is shown below:

$$r_t = w^{\mathrm{p}} r_t^{\mathrm{p}} + w^{\mathrm{v}} r_t^{\mathrm{v}} + w^{\mathrm{e}} r_t^{\mathrm{e}} + w^{\mathrm{rp}} r_t^{\mathrm{rp}} + w^{\mathrm{rv}} r_t^{\mathrm{rv}} \tag{1}$$

$$w^{\mathrm{p}} = 0.5,\ w^{\mathrm{v}} = 0.05,\ w^{\mathrm{e}} = 0.2,\ w^{\mathrm{rp}} = 0.15,\ w^{\mathrm{rv}} = 0.1$$

Where each reward r are pose, joint velocity, end-effector position, root pose, and root velocity rewards. w correspond to the weight of each reward. To train the policy, randomized ensembled double Q-learning (REDQ) [12], a model free off-policy algorithm is used. This algorithm handles overestimation from too many gradient steps by utilizing an ensemble of Q-functions. Standard dynamic randomization is used when training the baseline. Dynamic randomization indicates varying of mass, inertia, motor strength, friction, and latency.

Incorporating External Perturbations. Unlike previous works on quadruped locomotion, we incorporate another factor that is unique to guide dogs that can affect walking: the force coming from the human holding the rigid harness attached to the dog. We call the disturbing forces coming from the harness and other factors as *external perturbations*. In our pre-training process, not only does the robot learn to walk forward following the reference motions, but it also learns to resist external forces while walking. This is done by introducing external perturbations in the training process.

We experiment on two different types of perturbations: human gait perturbation and random push perturbation. The human gait perturbation is generated by the human gait as modeled in Section III-A. Random push perturbation applies forces of random (but capped) capacity and direction. During training, perturbations are applied to the harness (human gait) or to a random part of the robot body (random perturbation). In both cases, perturbations are applied in an interval.

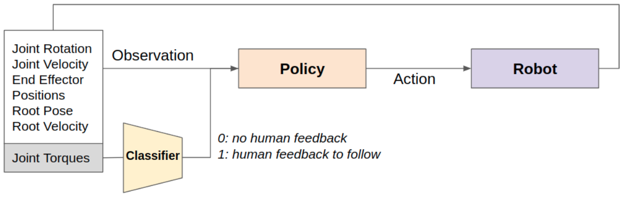

Fig. 2. The model architecture of the fine-tuning process. The classifier is an LSTM network with a linear layer attached at the end that distinguishes whether the input joint torque has human feedback force or not.

Three different variations (human gait, random push, human gait and random push) are trained and compared with the baseline, which is a locomotion policy trained without any perturbations. The results and analysis are detailed in Section IV. We use the best performing pre-trained policy (trained with only random push perturbations) in the fine-tuning process.

C. Fine-tuning to Follow Human Feedback

Unlike the reference paper [10] that further fine-tunes the locomotion policy in real world, we try to learn a different task in fine-tuning. In our approach, the purpose of fine-tuning is to learn to walk while following the feedback force given by the human through the harness. Unlike random perturbations such as forces applied during walking, a *feedback force* refers to forces with intention. Examples of feedback force can be pulling on the harness to make the guide dog slow down, or twisting the harness to nudge the dog in a different direction. In order to learn a policy that follows the human feedback, we employ rewards that minimize the feedback force. Our intuition is that to minimize the feedback force, the robot will have to move in the direction of the force. Fig. 2 shows the model architecture of the fine-tuning process. The pre-trained policy is carried over and further trained with the new reward and the output from the perturbation classifier.

$$r_t^{\mathrm{fine}} = r_t + w^{\mathrm{f}} r_t^{\mathrm{f}} + w^{\mathrm{h}} r_t^{\mathrm{h}} \tag{2}$$

$$r_t^{\mathrm{f}} = e^{-\lambda ||\tau^{\mathrm{ref}} - \tau^t||^2} \tag{3}$$

$$r_t^{\mathrm{h}} = e^{-\lambda ||h^{\mathrm{ref}} - h^t||^2} \tag{4}$$

$$w^{\mathrm{p}} = 0.45,\ w^{\mathrm{v}} = 0.04,\ w^{\mathrm{e}} = 0.2,\ w^{\mathrm{rp}} = 0.15,\ w^{\mathrm{rv}} = 0.1,$$
$$w^{\mathrm{f}} = 0.05,\ w^{\mathrm{h}} = 0.01$$

Equation 2 shows the reward used in the fine-tuning process. Two new rewards, the force reward r^{f} and torso height reward r^{h} is added to the existing reward r_t used in pre-training. The force reward r^f uses the joint torque from the robot and tries to minimize the additional torque caused by the human feedback force. In case of the human gait force, it will try to match the human gait. The torso height reward tries to minimize the difference between the current robot torso height and the reference height. It is a safeguard method to prevent the robot from collapsing and keep the robot level.

Test environment (Model: Pre-trained baseline)	Return
w/o perturbation	**337.1**
with perturbations	329.3

TABLE I
Comparison of the return for the pre-trained REDQ model with and without perturbations.

Model	Return
Baseline	169.3
Trained w/ random perturbations	**228.9**
Trained w/ human gait perturbations	204.2
Trained w/ random and human gait perturbations	221.1

TABLE II
Average return of pre-trained REDQ models with different perturbations. Random human gait perturbations are applied to the harness in testing.

Both rewards are calculated based on the L2 norm between the current value and the reference value. The reference value is the value (joint torque or torso height) before the feedback force is applied.

Perturbation Classifier. Ideally, the robotic guide dog should be able to walk stably, resisting external perturbations while following the human feedback. To enable this, the policy should be able to distinguish between the two, and take different actions depending the that type. For example, when external perturbation is detected, the policy should output action that allows the robot to walk stably *resisting* the applied force. On the other hand, if human feedback force is detected, the output action should *follow* the applied force.

In order to distinguish if the applied force to the robot is a human feedback force or a random perturbation, we introduce a perturbation classifier. A set window of joint torque history is used as the input to the classifier. Since the problem is formulated as timeseries classification, we choose to use an LSTM architecture [15]. In this paper, human gait force represents human feedback force. The LSTM perturbation classifier reads in the joint torques from the robot and returns 1 if human gait force is detected, and 0 otherwise. In this research we use a perturbation classifier separately trained with joint torques with a test accuracy of 98%.

The experimental results for fine-tuning are demonstrated in Section IV. We find out that this approach does not work as intended. We explain why our fine-tuning approach did not work and present an alternative approach in Section V.

IV. Results

In this section we present experiment results with the A1 robot in simulation. The experiments are designed to answer the following questions:

1) Does human gait perturbation have a significant effect in the policy's ability to walk stably?
2) What effect does learning with different types of perturbation (human gait, random) in pretraining have on the policy's ability to walk stably?
3) Can we finetune the pretrained policy to follow the direction of the applied force by introducing additional reward?

First, we conduct a preliminary experiment to justify pre-training with perturbations. Then, we compare the results of the locomotion model trained with and without human gait perturbations.

Preliminary Experiment. The provided pre-trained forward gait policy from the paper [10] is used to test if the trained policy can walk stably when human gait perturbation is applied. As noted in Table I, we find that the reward drops when human gait perturbation is applied indicating less stability when such perturbation is introduced. This drop justifies applying perturbations in the pre-training process. As a preliminary experiment, we only include human gait perturbation from one person.

Pre-training Locomotion. Table II displays the performance of pre-trained policies trained with different perturbations. All four policies are evaluated in a simulation environment with human gait perturbation applied to the harness. For fairness, human gait data different from the one used in training is used. The data shown are the average return of the policy rollout after 100 episodes. Higher return indicates that the robot follows the forward pace reference motion better. We can see that the policy trained with random perturbations has the highest return. Although tested on similar human gait perturbations, the policy trained with human gait forces performs worst among the three approaches. The policy trained with both random and human gait perturbations perform second-best, but the learned policy fails to advance forward when visually observed. We think the superior performance of the random perturbation approach is due to its generalizability compared to other perturbations. Not only can the robot resist human gait forces, but it can battle other forces such as the human bumping to the robot body.

Fine-tuning. Fig. 3 displays the visual results after fine-tuning with additional rewards. We explore various weight settings during the fine-tuning step. The weight setting presented in Sec. III-C was used to visualize Fig. 3. The robot was not able to walk and fell immediately, as shown in iteration 300 of Fig. 3. After 900 training iterations, the robot was able to avoid falling down, but was barely walking forward.

The rewards used in the pre-training and fine-tuning steps are designed to guide the policy to follow the reference motion. In the fine-tuning process, the force and torso rewards are added to follow the direction of the applied force. This results in one set of rewards trying to follow the reference motion, and another set of rewards trying to deviate from the reference motion. The adversarial nature of the two sets of rewards ultimately hinders the robot from learning. Such behavior is further demonstrated in Fig. 4, which shows the learning curve of the pre-training and fine-tuning policies. While pre-training learns to walk at around 200 iterations, fine-tuning learns slowly and has comparatively low returns even after

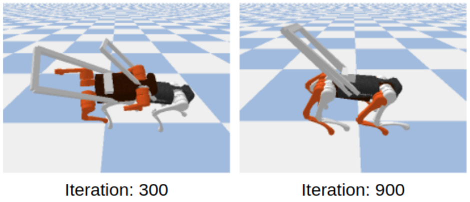

Fig. 3. Visual results of the policy after fine tuning for 300 and 900 iterations.

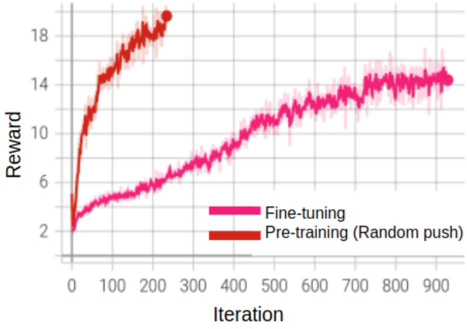

Fig. 4. Learning curves of training reward of the pre-training and fine-tuning process.

900 iterations.

We think incorporating adversarial rewards as a reward function is not an ideal approach. An alternative approach that potentially avoids this problem is explained in Section V.

V. Conclusion

In this project, we tackle the problem of teaching a robotic guide dog to walk. We try to tackle two main tasks: walking while resisting external forces, and following the human feedback coming from the harness. For the first task, we observed that pre-training the motion imitation model with random perturbations is effective when human gait perturbation is applied in testing. The pre-trained policy with perturbation performed better in following the ground truth motions compared to the baseline policy.

However, our approach for following human feedback by fine-tuning with additional rewards did not work as expected. We believe this is mainly due to the adversarial nature of the existing rewards and the additional rewards. To resolve this, we propose another approach that uses the output from the pre-trained policy and the perturbation classifier to train a separate policy network with the force and torso rewards. The model architecture is illustrated in Fig. 5. We think this is a promising direction because we can avoid the clashing reward problem from the previous approach, while maintaining knowledge of basic locomotion from the pre-trained policy. We hope to achieve our goal for fine-tuning by experimenting on this new approach.

In the future, we plan to test our model on a real robot. Assuming both the pre-training and fine-tuning process works in simulation, we think we can achieve similar results in the real world with a bit of further learning. Also, rather than using human gait forces for the feedback force, we plan to use actual feedback forces such as pushing, pulling, and twisting the harness in fine-tuning.

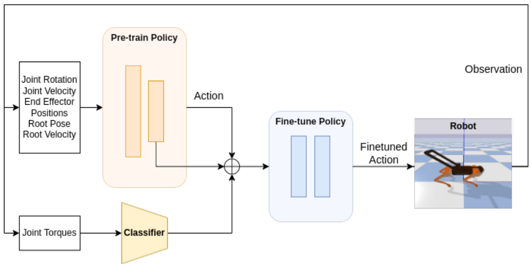

Fig. 5. Proposed alternative fine-tuning approach that trains a new locomotion policy with the feature vector from the pre-trained policy and the perturbation classifier.

References

[1] E. E. Bray, M. D. Sammel, D. L. Cheney, J. A. Serpell, and R. M. Seyfarth, "Effects of maternal investment, temperament, and cognition on guide dog success," *Proceedings of the National Academy of Sciences*, vol. 114, no. 34, pp. 9128–9133, 2017.

[2] L. Wang, J. Zhao, and L. Zhang, "Navdog: robotic navigation guide dog via model predictive control and human-robot modeling," in *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, 2021, pp. 815–818.

[3] Y. Wei, X. Kou, and M. C. Lee, "Smart rope and vision based guide-dog robot system for the visually impaired self-walking in urban system," in *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2013, pp. 698–703.

[4] H. Tan, C. Chen, X. Luo, J. Zhang, C. Seibold, K. Yang, and R. Stiefelhagen, "Flying guide dog: Walkable path discovery for the visually impaired utilizing drones and transformer-based semantic segmentation," in *2021 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2021.

[5] T.-K. Chuang, N.-C. Lin, J.-S. Chen, C.-H. Hung, Y.-W. Huang, C. Teng, H. Huang, L.-F. Yu, L. Giarré, and H.-C. Wang, "Deep trail-following robotic guide dog in pedestrian environments for people who are blind and visually impaired-learning from virtual and real worlds," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 5849–5855.

[6] J. Wilson, B. N. Walker, J. Lindsay, C. Cambias, and F. Dellaert, "Swan: System for wearable audio navigation," in *2007 11th IEEE international symposium on wearable computers*. IEEE, 2007, pp. 91–98.

[7] J. Tan, T. Zhang, E. Coumans, A. Iscen, Y. Bai, D. Hafner, S. Bohez, and V. Vanhoucke, "Sim-to-real: Learning agile locomotion for quadruped robots," *arXiv preprint arXiv:1804.10332*, 2018.

[8] T. Haarnoja, S. Ha, A. Zhou, J. Tan, G. Tucker, and S. Levine, "Learning to walk via deep reinforcement learning," *arXiv preprint arXiv:1812.11103*, 2018.

[9] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science robotics*, vol. 5, no. 47, p. eabc5986, 2020.

[10] L. Smith, J. C. Kew, X. B. Peng, S. Ha, J. Tan, and S. Levine, "Legged robots that keep on learning: Fine-tuning locomotion policies in the real world," 2021.

[11] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "Rma: Rapid motor adaptation for legged robots," *arXiv preprint arXiv:2107.04034*, 2021.

[12] X. Chen, C. Wang, Z. Zhou, and K. Ross, "Randomized ensembled double q-learning: Learning fast without a model," *arXiv preprint arXiv:2101.05982*, 2021.

[13] J. Camargo, A. Ramanathan, W. Flanagan, and A. Young, "A comprehensive, open-source dataset of lower limb biomechanics in multiple conditions of stairs, ramps, and level-ground ambulation and transitions," *Journal of Biomechanics*, vol. 119, p. 110320, 2021.

[14] X. B. Peng, E. Coumans, T. Zhang, T.-W. E. Lee, J. Tan, and S. Levine, "Learning agile robotic locomotion skills by imitating animals," in *Robotics: Science and Systems*, 07 2020.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.