A proof-of-concept framework for adaptive haptic feedback via reinforcement learning

Zachary Logan, Sophie Walsh, Quentin Anderson-Watson, Katie Fitzsimons

***Abstract*— Haptic wearables are capable of enhancing user experiences, reducing collisions, and improving virtual training in robotic and VR applications. However, past studies have shown that contradictory evidence on which feedback paradigm is the most effective, leading to many using handcrafted solutions. We hypothesize that reinforcement learning can be leveraged to adapt a haptic feedback policy in real time to account for individual differences in user response. In this study, we trained a neural network–based policy using proximal policy optimization to provide instructions via a haptic display to participants as they moved a joystick and evaluated the feasibility of adapting the policy to individual differences in response. The results of this study demonstrate the feasibility of this method of policy adaptation on wearable haptic devices, and indicate that the policies generated differ between individuals. This work serves as a proof-of-concept for an experimental framework to study adaptive haptic feedback in human-in-the-loop settings.**

I. Introduction

Wearable haptic interfaces provide a channel for autonomous systems to communicate information to humans without competing with visual or auditory stimuli. Vibrotactile feedback has been shown to enhance interaction across diverse domains: improving consumer device usability [1], supporting spatial navigation [2], substituting kinesthetic feedback [3], reducing powered wheelchair collisions [4], assisting drive-by-wire platforms [5], enhancing motor control [6], and enriching virtual training [7].

Despite this promise, robust implementation remains challenging. Human haptic perception is context-dependent and can be degraded by several well-documented phenomena. Tactile suppression—the reduced perception of haptic cues during movement—emerges across tasks ranging from simple finger motions [8], [9] to whole-arm actions [10], [11]. Tactile masking occurs when simultaneous stimuli interfere with one another, as in the suppression of skin-stretch cues by skin-squeeze cues [12]. Beyond these physiological effects, user interpretation is further shaped by cognitive load [13], [14], intuitiveness of cue mappings [15], and individual preferences [16], [10], [17], [2]. As a result, even with perfect perception, effective action on haptic cues often requires extensive training or highly intuitive encodings [18].

Most existing vibrotactile paradigms employ fixed mappings (e.g., encoding error as vibration intensity). These schemes are easy to implement but yield inconsistent results: some studies find repulsive encodings more effective [3], [16], while others favor attractive mappings [2]. Variability across users likely reflects both perceptual differences and interface-specific perturbations, such as actuator placement or contact force. Data-driven methods have attempted to address this by fitting offline regression models that rescale cues based on sensitivity [19], [20]. While these approaches reduce perceptual errors, they assume static encodings, cannot adapt online, and fail to capture higher-level interpretation differences.

Parameterized perceptual models could be updated incrementally using gradient-based methods rather than relying on fixed encodings, but this would allow for only limited personalization as they are typically simple linear regression-based models. As a result, incremental updates are unlikely to capture complex co-adaptive behaviors that arise when both the user and the system change over time.

Reinforcement learning offers a natural framework for addressing nonlinear, co-adaptive interaction because it updates policies based on trial-by-trial evaluative feedback rather than requiring predefined supervisory targets for each stimulus–response pair. However, not all reinforcement learning methods are appropriate for human-in-the-loop settings. While direct policy optimization and bandit-based methods are capable of responding to changes in the environment in real time, they permit unconstrained parameter updates. For example, policy gradient methods such as REINFORCE [21], Actor-Critic [22] and Asynchronous Advantage Actor–Critic [23] can perform online updates to their policies in real-time; however, they do not impose any constraints on the magnitude of policy updates. making them less suitable for human-in-the-loop systems where feedback stability is critical. Similarly bandit algorithms such as, ε-greedy [24], Upper Confidence Bounds [25], and Bayesian bandits [26] update action-selection strategies online but allow abrupt shifts in behavior following individual reward observations. In human-in-the-loop systems, such unconstrained updates can cause sudden and unintelligible changes in feedback, destabilizing human learning and increasing response variability. The limitations imposed by unconstrained updates directly motivated the development of trust-region policy methods, which explicitly constrain how much a policy may change between updates in order to ensure stable and predictable behavior during learning.

Trust-region policy methods were developed to address instability caused by large policy updates by explicitly constraining how much a policy may change between iterations. Trust Region Policy Optimization (TRPO) [27], enforces a strict divergence constraint but relies on second-order opti-

Zachary Logan, Sophie Walsh, Quentin Anderson-Watson, Katie Fitzsimons are with Department of Mechanical Engineering, The Pennsylvania State University, University Park, PA 16802, USA zal5@psu.edu, sfw5702@psu.edu, qxa5031@psu.edu, k-fitzsimons@psu.edu

mization and line search procedures, introducing substantial computational overhead. Such overhead limits practicality for real-time human-in-the-loop learning. Proximal Policy Optimization (PPO) [28], approximates trust-region behavior using a clipped objective function, enabling stable and incremental policy updates while remaining computationally efficient. Therefore, we propose an online adaptation framework for haptic feedback using PPO. Specifically, we outline a method for online adaptation of haptic feedback using proximal policy optimization. We empirically demonstrate that this method can converge in a reasonable time frame to a unique policy for each user. Such individualized adaptation may be particularly valuable in human–robot interaction scenarios where haptic feedback conveys robot state or navigation information to a human operator, including teleoperation systems and upper-limb prosthetic devices in which vibrotactile cues are used to communicate task-relevant information. In these contexts, user-specific perceptual differences can significantly influence feedback interpretability, motivating adaptive personalization strategies.

II. Methods

In this study, we developed and evaluated an online learning framework that adapts a haptic feedback mapping to individual users and perturbations using proximal policy optimization (PPO), as shown in Figure 1. To accomplish this, we used a pretraining step to learn an initial policy that represents a generic haptic feedback mapping. We defined a reward function for online training and evaluated the online adaptation framework in a study with five participants. We assessed the convergence of the online adaptation framework across users and its uniqueness.

A. Policy Pre-training

Human-in-the-loop trials are inherently costly and constrained, as each trial requires sustained user attention, active interpretation of haptic cues, and repeated motor responses. Extended sessions are further limited by mental fatigue, degradation of local sensitivity from repeated exposure, and learning effects, all of which can degrade the quality of the data. To reduce the number of human-in-the-loop trials needed for convergence, we pre-trained a multilayer perceptron (MLP) using a simple mapping from the desired joystick direction to the intensities of the haptic feedback motors. Given that we have a four-dimensional continuous action space, and the requirement to have an interpretable and spatially coherent haptic patterns, successfully learning this encoding through uninformed exploration could take upwards of thousands of human interaction trials—far exceeding what can be reasonably be obtained from a participant in a single session.

The goal of this pre-training step was not to generate an initial model of human behavior and perception, but to encode a simple haptic feedback policy based on a linear mapping between vibration intensity and perceived stimuli location. Alternative surrogate datasets could include simulated user responses or data collected from users who participated in similar studies; however, both approaches would require the introduction of some assumptions on perception, interpretation strategies, motor responses that would be highly user-specific and not generalizable. Whereas the chosen synthetic dataset used here encodes only the geometrical relationship between which tactors would correspond to which direction without embedding any assumptions about user interpretations or learning. This setup allows the pre-training stage to generate a spatially coherent and simple interpretable mapping that can be adapted to each individual user.

The mapping assumes a linear relationship between the relative intensities of two adjacent tactors to generate a phantom sensation between the two. The MLP was initialized as a fully connected, feedforward neural network with one input node, two hidden layers of 64 nodes each, and four output nodes, using the ReLU activation function. The input was the desired joystick directions normalized to be between -1 and 1, and the four output nodes were the mean intensity for each of the four vibration motors, normalized from 0 to 1. We generated the synthetic training data set composed of 5000 data points by mapping the desired joystick direction to a generalized set of motor intensities, $\mathbf{I}_d$, using equation 1, where ϕ_d corresponds to the desired directions of joystick movement.

Equation 1, is a simple linear mapping based on the phantom sensation haptic phenomenon [29], where each joystick direction ϕ_d, is represented by activating either a single tactor or a pair of tactors.

$$\mathbf{I}_d(\phi_d) = \begin{cases} [0.0, 0.0, 0.0, 0.0] & \phi_d = 0 \\ [1.0, 0.0, 0.0, 0.0] & \phi_d = 1 \\ [0.5, 0.5, 0.0, 0.0] & \phi_d = 2 \\ [0.0, 1.0, 0.0, 0.0] & \phi_d = 3 \\ [0.0, 0.5, 0.5, 0.0] & \phi_d = 4 \\ [0.0, 0.0, 1.0, 0.0] & \phi_d = 5 \\ [0.0, 0.0, 0.5, 0.5] & \phi_d = 6 \\ [0.0, 0.0, 0.0, 1.0] & \phi_d = 7 \\ [0.5, 0.0, 0.0, 0.5] & \phi_d = 8 \end{cases} \quad (1)$$

For initialization, the mapping is specified for nine discrete target directions; however, the underlying motor control space remains continuous, and intermediate directions can be represented through arbitrary intensity combinations. When a desired direction is aligned with a motor, the activation of that motor is maximum. For directions between motors, two adjacent motors are activated such that the level of activation is proportional to their distance from the desired direction. For example a direction halfway between two tactors would be displayed by 50% intensity of the two adjacent tactors to generate a phantom point of vibration between the two tactors. This equation reflects previous work in tactile perception that found that relative intensity across adjacent actuators is interpreted as a continuous spatial cue rather than two separate points and provides a simple baseline mapping between joystick direction and perceived

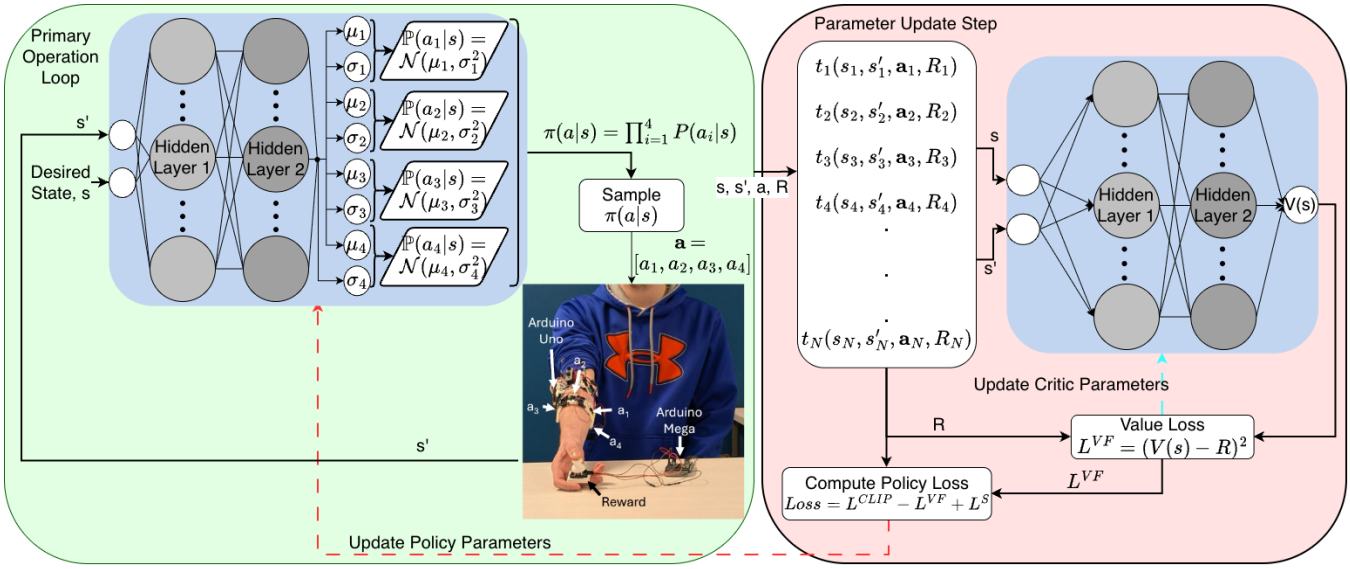

Fig. 1. Overview of the PPO-based training loop for the haptic feedback policy. At the beginning of each trial, the current policy receives a desired direction for the joystick and outputs motor commands corresponding to the vibration intensity. These commands are applied as haptic stimuli via the tactors to the participant's arm. The participant interprets the haptic stimuli as instructions and moves the joystick toward the most intense vibration. Their response is recorded and compared to the desired direction to compute a reward value. This response, the original desired direction, and the current policy distribution are used to calculate the PPO loss and update the policy parameters. The process repeats across trials to refine the haptic feedback policy based on the user's behavior.

haptic direction.

The MLP was trained on this data using the mean squared error loss between the predicted and actual motor intensities supplied by our mathematical mapping, as shown in equation 2, with a learning rate of 0.0003, 1000 epochs, and the ADAM optimizer.

$$Loss = \frac{1}{n} * \sum \mathbf{I}_{actual} - \mathbf{I}_d \tag{2}$$

Training of the MLP continued until either the MLP had finished 5000 epochs or until the error between the predicted motor intensities and the actual motor intensities was within a tolerance of 1×10^{-5}. After training was complete, the MLP's weights and biases were used to initialize the policy for online adaptation of the haptic feedback.

B. Online Learning Framework

We selected the clipped object variant of PPO [28], implemented using the Advantage Actor-Critic framework [30], [31], to explicitly limit how much the magnitude the policy can change during each update. This limitation ensures gradual, stable changes in haptic feedback that preserves interpretability while still allowing adaptation to user-specific perception and evolving task dynamics.

In this formulation, the interaction is modeled as an approximate Markov Decision Process. At each trial t, the policy receives the desired joystick direction, ϕ_d, and outputs a continuous four-dimensional vector of motor intensities as the action. The true environment state, however, includes not only the externally specified direction but also the participant's internal perceptual mapping between vibration patterns and interpreted direction, which is not directly observable and may evolve over time. As a result, the interaction is more accurately characterized as a partially observable and potentially nonstationary process. We nevertheless employ PPO due to its empirical robustness to partial observability and changing dynamics, and its ability to optimize perception–action coupling over batches of data rather than relying on locally linear, pointwise adaptation. The reward at trial t therefore depends on both the current action and prior policy updates that have influenced user interpretation.

The PPO policy was initialized with the weights and biases from the pre-trained MLP, and the PPO value function was initialized with randomly generated weights and biases using the Stable Baselines3 library [31]. To limit rapid deviation from the pre-trained policy, we set a small learning rate of 0.0001, which constrained the magnitude of the policy updates. We also initialized the logarithmic standard deviation matrix of the policy's action distribution to values between -5 and -1; preliminary testing showed that these values preserved the pre-trained mapping during early training while allowing limited exploration. To encourage exploration around the pre-trained policy's mean actions—without overwhelming the pretrained behavior—we set the entropy coefficient, c_2, to 0.01. Due to limited data, we set the batch size to 10 to ensure adequate sample efficiency and the number of epochs to 15 to balance training performance and the risk of overfitting. The value function coefficient, c_1 was set to 0.0 and the clipping ratio, ε, to 0.2 based on the typical values used in previous PPO implementations [28], [32].

In our experimental setup, each trial yields a single user response, resulting in sparse human-generated rewards.

We designed a piecewise reward function based on the normalized absolute value of the wrapped direction error (equation 3). In equation 4, when the absolute value of the normalized error is equal to zero, the policy receives a reward of $+1$ for that action, which helps increase the rate at which desirable actions are attributed to the correct joystick directions, ϕ_d. For non-zero errors, the reward is set to -2 times the error magnitude, ρ_ϕ. This results in a penalty range of -1 to 0, which aligns with the normalized scales used for joystick input and motor output.

$$\rho_\phi = \frac{\min(|\phi_d - \phi_u|, 8 - |\phi_d - \phi_u|)}{8} \tag{3}$$

$$Reward = \begin{cases} +1 & \text{if } \rho_\phi = 0 \\ -2 * \rho_\phi & \text{if } \rho_\phi \neq 0 \end{cases} \tag{4}$$

C. Experimental Platform

We used a custom open-source haptic platform, the Snaptics system [33]. This low-cost open-source wearable device is designed to deliver tactile feedback to the skin. The platform consisted of two armbands worn on the forearm. The first armband contained four eccentric rotating mass (ERM) mini-motors (MicroDCMotors), each measuring 4 mm in diameter and 11 mm in length, and operating at 183 Hz under a 3V input. These motors were arranged equidistantly around the circumference of the forearm in a circular layout, similar to previous designs that leverage phantom sensations [34], [35], [36], as illustrated in Figure 2. The second armband housed the control electronics: two DRV8883 motor drivers and an Arduino Uno microcontroller. This microcontroller communicated with a host computer to synchronize haptic cues with collected kinematic data during the trials. Both armbands were secured to the participant's right forearm using adjustable nylon straps. The four tactors were positioned at the cardinal directions around the forearm, and intermediate joystick directions were represented through relative intensity modulation across adjacent actuators.

We utilized an analog gaming joystick, similar to those in video game controllers sourced from Elegoo, as shown in Figure 2. The joystick contains two potentiometers that produce analog electrical signals corresponding to the joystick's horizontal and vertical displacement. During trials, participants sat in a chair, held the joystick in their right hand, and moved the joystick with their thumb. An Arduino Mega was used to read the two analog signals and transmit them to the host computer, where the joystick's directional input and displacement from center were computed.

D. Experimental Protocol

Five participants were recruited for this study and provided informed written consent before participation. The protocol for this research study was reviewed and approved by the Penn State Institutional Review Board under STUDY00026482. Each participant completed a one-hour session consisting of 270 trials, which included 30 repetitions of the nine possible joystick directions. Upon enrollment,

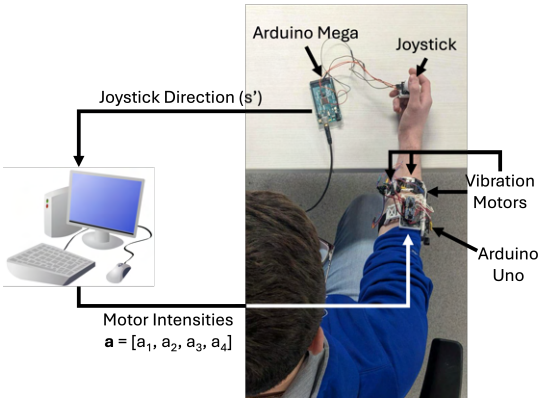

Fig. 2. The experimental platform consisted of a haptic armband, an analog joystick, and a host computer. The haptic platform featured four eccentric rotating mass (ERM) mini-motors arranged in a circular array around the participant's forearm, aligned with the four cardinal directions (Up, Down, Left, and Right). These motors were controlled by an Arduino Uno connected to the host computer. The analog joystick was a two-axis joystick from Elegoo, which measured thumb movements via two potentiometers mounted on a gimbal mechanism. The joystick signals were read by an Arduino Mega and transmitted to the host computer. During each trial, the host computer sent vibration intensity commands to the Arduino Uno, prompting the motors to activate. Participants interpreted the resulting haptic cues as directional instructions and moved the joystick accordingly. The Arduino Mega then relayed the joystick position back to the host computer for evaluation.

participants are assigned a randomized lists of joystick directions each was generated using a random number generator to ensure even sampling across directions.

Prior to the online learning session, participants completed 10 practice trials using the pre-trained policy to familiarize themselves with the system. Immediately following these practice trials, participants completed a subjective usability questionnaire assessing their experience with the baseline feedback condition. The questionnaire included items from the System Usability Scale (SUS) [37], along with three open-ended questions. The open-ended prompts asked participants: (1) whether they preferred any particular version or presentation of the feedback, (2) what, if anything, they liked about the feedback, and (3) what, if anything, they disliked about the feedback.

After completion of the baseline survey, we initialized the PPO policy using a pre-trained multilayer perceptron. During each trial, participants were instructed to move the joystick in the direction indicated by the haptic feedback platform, corresponding to where they felt the most intense vibration. After each trial, we passed the collected data to the proximal policy optimization algorithm to evaluate the performance of the current haptic feedback mapping and optimize the mapping by performing a policy update. Following the completion of the 270 trials, participants were presented with the same subjective usability questionnaire, evaluating the adapted, user-specific policy.

E. Data Analysis

Policy convergence was assessed by computing the policy loss rate by applying a moving-average slope method to the

policy loss and fitting a linear model to the last 10 trials. A policy was considered converged if the absolute value of this slope was below 0.005 or if the change was less than 0.5%, indicating that the loss had stabilized.

Policy performance was quantified using directional error, defined as the absolute difference between the instructed direction and the participant's joystick response (wrapped to the minimum circular distance). Directional error was averaged within sliding windows across trials to assess trends during learning. A decrease in mean directional error over time was interpreted as evidence that the adapted policy improved the interpretability of the haptic encoding.

Subjective usability was quantified using the System Usability Survey (SUS) [37]. Individual item responses were scored according to standard SUS procedures and converted to a composite score ranging from 0–100. Mean SUS scores were computed for each feedback condition. Paired statistical comparisons were performed to evaluate whether the learned, user-specific policy differed from the baseline policy in perceived usability. Open-ended responses were analyzed using qualitative thematic analysis. Responses were reviewed and coded to identify recurring themes related to clarity, comfort, interpretability, and perceived differences between feedback versions.

A policy uniqueness metric was quantified by computing the Jensen-Shannon divergence (JSD) [38] between the action space distributions of each participant's learned policy and the pre-trained baseline policy. For each actuator, we computed the JSD between the two policies' action distributions at each of the nine joystick directions. These nine JSD values were then averaged to obtain that actuator's Average JSD. This procedure was repeated for all four actuators. The overall Average JSD for a participant was computed as the mean of the four actuator-specific Average JSD values. This measure reflects how distinct or unique a learned policy for an individual participant is compared to the baseline pre-trained policy. To evaluate whether these JSD values differed significantly, a one-way analysis of variance (ANOVA) was performed with the participant as the factor. A significant result would indicate that the degree to which participant-specific policies diverged from baseline varied by participant. To further examine pairwise differences, a Student's paired t-test was performed on the JSD values between each possible pairing of participants. The null hypothesis for each comparison was that there was no significant difference in the JSDs of the two participants at baseline. Rejecting the null hypothesis would indicate that participants developed policies that diverged from the baseline in ways specific to each participant, supporting the idea that policy learning was individualized. Post Hoc corrections were performed with the Benjamini-Hochberg [39] false discovery rate method to adjust the p-values.

III. Results

To evaluate the convergence of participant policies in the PPO training process, we analyzed the loss rate of the policy loss over the last 8 trials for each participant, as shown in Table I. The final loss rate for the policies of participants 1-4, ranged from 0.0001 to 0.0031, which are all below the threshold, suggesting that the policies have successfully converged to a solution. However, the final loss rate for participant 5 was 0.0071, which is well above threshold, indicating significant variability between updates and that the policy had not fully converged.

Participant	Final Loss Slope
1	0.0021
2	0.0031
3	0.0023
4	0.0001
5	0.0071

TABLE I

ABSOLUTE VALUE OF THE FINAL POLICY LOSS RATE FOR PARTICIPANTS 1-5, COMPUTED OVER THE LAST 10 TRIALS. LOSS RATES WITH ABSOLUTE VALUES BELOW THE THRESHOLD OF 0.005 SUGGEST SUCCESSFUL POLICY CONVERGENCE. THE LOSS RATES FOR THE POLICIES OF PARTICIPANTS 1-4 WERE BELOW THE THRESHOLD, INDICATING SUCCESSFUL CONVERGENCE, AND THE LOSS RATE FOR PARTICIPANT 5 WAS ABOVE THE THRESHOLD INDICATING THE POLICY HAD NOT FULLY CONVERGED.

Directional error was analyzed using non-overlapping sliding window averages across trials. Initial window means ranged from 0.4 to 2.8 across participants, while final window means ranged from 0.6 to 1.0. Absolute changes between initial and final windows varied by participant, with reductions observed for participants 1 (2.0), 2 (0.4), and 4 (1.4), no change for participant 5 (0.0), and an increase for participant 3 (-0.6). Early versus late window comparisons were not statistically significant for any participant Overall, no statistically significant changes in directional error were observed over the course of training.

Survey responses and task error metrics were additionally analyzed to assess changes across conditions. Mean survey scores were $\mu = 78.75$ *and* $SD = 7.77$ at baseline and $\mu = 74.38$ *and* $SD = 6.57$ following personalization. A paired t-test indicated no significant difference between conditions ($t = 1.331,\ p = 0.275$).

Table II summarizes the Jensen–Shannon Divergence (JSD) between each participant's learned policy and our pre-trained policy. Average JSD values varied across participants, with 1 and 2 showing the lowest divergence (0.1336 and 0.1318, respectively), while participants 3 and 4 exhibited higher divergence from the pretrained policy (0.2479 and 0.2117). To examine the impact the participant had on the amount of divergence from the pre-trained policy, we ran a Shapiro-Wilk and Levene's test over the JSD values to check the required assumptions to run an ANOVA. The Shapiro-Wilk test showed that the JSD values were approximately normal ($W = 0.917,\ p = 0.088$), and the Levene's test indicated that the data had homogeneity of variance across all groups ($F = 1.256,\ p = 0.330$). With the assumptions of

the ANOVA satisfied, a one-way ANOVA, with participant as the factor, was run over the JSD values. The analysis showed that the effect of participant ($F = 0.886,\ p = 0.496$) was not statistically significant. These results indicate that the average divergence from the pre-trained policy did not significantly differ across participants. However, the mean JSD values were greater than zero for all participants, indicating that each learned policy deviated from the pretrained initialization.

Participant	Motor 1	Motor 2	Motor 3	Motor 4	Avg. JSD
1	0.1406	0.0616	0.2468	0.0854	0.1336
2	0.1430	0.0926	0.0524	0.2391	0.1318
3	0.4569	0.2779	0.0648	0.1920	0.2479
4	0.1216	0.2639	0.2565	0.2051	0.2117
5	0.0817	0.2887	0.2987	0.1107	0.1950

TABLE II

AVERAGE JENSEN-SHANNON DIVERGENCE (JSD) CALCULATED BETWEEN THE COMPARISON BETWEEN EACH PARTICIPANT ACROSS MOTORS, INCLUDING OVERALL TOTAL AVERAGE JSD ACROSS ALL MOTORS AND DIRECTIONS WITH THE PRE-TRAINED POLICY.

While these analyses assess how much each participant diverged from the pre-trained policy, they do not capture how participants' learned policies differed from one another. To evaluate this, JSD was computed directly between all pairs of participant policies. Table III and Figure 3 present the JSD between the learned policies of all participant pairs. Average JSD values varied across pairs. Some pairs, such as participants 1 and 2, had a relatively low JSD ($Average\ JSD = 0.1784$), showed only small difference in policy behavior, while other pairs, such as participants 4 and 5 ($Average\ JSD = 0.3463$), exhibited a relatiely large JSD, which indicates more distinct policy behavior learned from those two users.

One-sample t-tests conducted for each participant pair demonstrated that all participant-to-participant divergences were significantly greater than zero after Benjamini–Hochberg correction (all $p < 0.001$), as shown in Table III. This indicates that every participant's learned policy was statistically different from every other participant's policy.

IV. DISCUSSION

Prior research shows that user performance differs even for simple motions when different haptic encodings are used. For instance, the effectiveness of haptic cues is sometimes better when repulsive cues are used [10], [3], while other studies demonstrate attractive [2] cues are better. This may be attributed to the fact that some users prefer one mapping over the other [40]. However, many haptic feedback systems rely on fixed, hand-crafted mappings, limiting their ability to adapt to different users and environments. Our results demonstrate that proximal policy optimization can learn haptic mappings that adapt to individual users.

Pair	Motor 1	Motor 2	Motor 3	Motor 4	Avg. JSD	p-value
1–2	0.0818	0.1060	0.2524	0.2733	0.1784	5.64×10^{-7}
1–3	0.5147	0.2702	0.2672	0.2233	0.3189	4.11×10^{-10}
1–4	0.1848	0.2726	0.4824	0.2221	0.2905	1.03×10^{-10}
1–5	0.1804	0.2927	0.3222	0.1077	0.2257	1.03×10^{-10}
2–3	0.5223	0.3509	0.0327	0.2841	0.2975	9.12×10^{-10}
2–4	0.1539	0.3376	0.3002	0.4042	0.2990	1.07×10^{-11}
2–5	0.1917	0.2111	0.2521	0.2991	0.2385	1.80×10^{-12}
3–4	0.4132	0.2160	0.2976	0.1378	0.2661	4.50×10^{-10}
3–5	0.4047	0.4893	0.2497	0.2364	0.3450	1.68×10^{-12}
4–5	0.1671	0.4684	0.4983	0.2517	0.3463	1.07×10^{-11}

TABLE III

AVERAGE JENSEN–SHANNON DIVERGENCE (JSD) BETWEEN EACH PAIR OF PARTICIPANTS' LEARNED POLICIES ACROSS ALL FOUR MOTORS. THE TABLE SHOWS BOTH MOTOR-SPECIFIC JSD VALUES, THE OVERALL AVERAGE JSD AND THE PAIRWISE T-TEST RESULT FOR EACH PARTICIPANT PAIR.

The policies for participants 1-4 successfully converged, indicating that this approach is feasible; however, the policy for participant 5 did not converge. This lack of convergence could be due to a variety of factors, such as initial overfitting, extremely low or high variability in human response, the selection of hyperparameters, or insufficient trials. These results suggest that policy convergence also depends on individual factors. While our selection of hyperparameters and training time worked for most participants, some users may require different parameters or additional training. Successful online learning depended on the availability of a reasonable pretrained initialization. The pretrained policy provided a stable starting point that allowed human-in-the-loop adaptation to proceed efficiently. This suggests that while reinforcement learning can personalize haptic mappings online, careful initialization remains a critical component of practical deployment.

Although policy convergence was observed among most participants, this adaptation did not result in statistically significant changes in directional error during training. Directional error remained stable across trials, and subjective usability survey scores did not differ significantly between pre- and post-personalization. These results suggest that personalization did not degrade task performance or perceived usability, nor did it produce measurable improvements within the scope of this task. The simplistic nature of the task

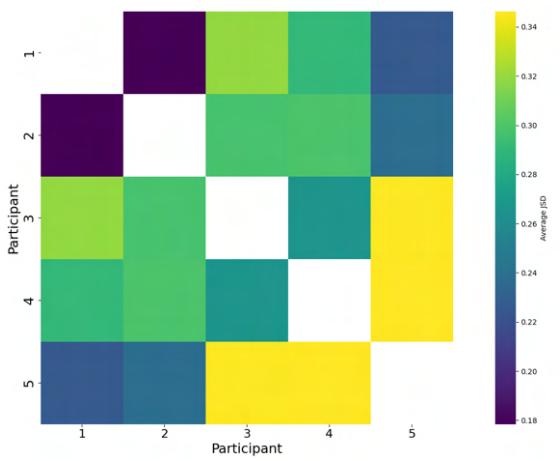

Fig. 3. Average Jensen–Shannon Divergence (JSD) between each pair of participants' learned policies across all four motors. Each cell in the matrix represents the overall JSD for the corresponding participant pair, with lighter colors indicating lower divergence and darker colors indicating higher divergence.

may have had a ceiling effect on performance improvements with personalization. Additionally, participants operated in a controlled setting with explicit knowledge of the feedback structure. In more ecological scenarios—such as naive users, more complex control spaces, prolonged exposure, or less intuitive feedback mappings—greater performance gains may emerge, particularly in situations where perceptual ambiguity is higher or users have not been preconditioned to interpret the feedback.

When comparing each participant's learned policy to the pretrained initialization, no statistically significant differences were found in the magnitude of divergence across participants. Although numerical differences in average JSD were observed, the ANOVA and pairwise tests indicated that the overall degree of deviation from the pretrained policy was comparable across individuals. This suggests that, in terms of overall deviation from initialization, participants required similar magnitudes of adaptation.

To examine whether the learned policies differed across participants, we computed pairwise JSD values between each participant's personalized policy. While divergence from the pretrained policy was statistically comparable across participants, this measure does not capture differences between their personalized policies. Our results for the pairwise JSD revealed that the personalized policies developed unique structures. For example, participants 1 and 2 exhibited a low pairwise JSD value, indicating that their learned policies were comparable. In contrast, participants 4 and 5 showed a substantially larger pairwise JSD, suggesting that their learned mappings differed considerably. These pairwise differences confirm that the learned policies reflected individualized adaptations rather than uniform deviations from the pretrained initialization.

Although these individualized adaptations did not correspond to measurable improvements in task accuracy, their presence indicates that participants did not converge to a single shared mapping; instead, the reinforcement learning framework adapted the feedback policy in a participant-specific manner. This finding highlights the potential value of personalized adaptation in haptic feedback systems, particularly for more complex tasks or feedback mappings that are less intuitive.

Generalized haptic feedback policies are attractive from a deployment perspective, as they require minimal calibration and exhibit predictable behavior across users. However, such fixed mappings may not account for individual perceptual differences. In contrast, individualized policies can adapt to each user's interpretation of the cues, potentially improving robustness to perceptual variability. This personalization introduces additional complexity and requires careful initialization to avoid overfitting, but our results suggest that reinforcement learning provides a viable framework for achieving this balance.

Overall, these findings demonstrate that proximal policy optimization can converge to stable, individualized haptic mappings in a human-in-the-loop setting without degrading task performance or usability. Although these results did not yield measurable performance gains, the emergence of distinct user-specific policies supports the feasibility of reinforcement learning-based personalization. This framework provides a promising foundation for developing adaptive wearable haptic systems in more complex or challenging human-robot interactions.

Acknowledgments

This material is based upon work supported by the U.S. National Science Foundation under Award No. 2319139. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author and do not necessarily reflect the views of the U.S. National Science Foundation.

V. Conclusion

This paper investigates the use of proximal policy optimization to implement a framework for the online adaptation of a haptic feedback policy. Feedback was conveyed using an array of four vibrotactile motors to indicate the desired direction for joystick movement. Our results demonstrate that proximal policy optimization can be implemented in a human-in-the-loop wearable system and can produce individualized feedback policies across users without degrading task performance. Although measurable performance gains were not observed in this simplified task, the emergence of distinct user-specific policies highlights the potential importance of personalized feedback, particularly for more complex or less intuitive interaction scenarios.

Future work will explore more sophisticated reward functions, various data acquisition strategies to improve the quality of the learned solutions, explore how a learned policy adapts to changes in perception or presentation of the cues, and examine how to incorporate subjective user preference measures to evaluate perceived comfort and intuitiveness. Additionally, future experiments will compare

different pretrained policy initializations (e.g., attractive- versus repulsive-based mappings) and assess performance differences between participants with and without prior exposure to the feedback structure.

REFERENCES

[1] T. Singhal and O. Schneider, "Juicy haptic design: Vibrotactile embellishments can improve player experience in games," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, (New York, NY, USA), Association for Computing Machinery, 2021.

[2] S. Günther, F. Müller, M. Funk, J. Kirchner, N. Dezfuli, and M. Mühlhäuser, "Tactileglove: Assistive spatial guidance in 3d space through vibrotactile navigation," in *Proceedings of the 11th pervasive technologies related to assistive environments conference*, (New York, NY, USA), pp. 273—-280, Association for Computing Machinery, 2018.

[3] P. Kapur, M. Jensen, L. J. Buxbaum, S. A. Jax, and K. J. Kuchenbecker, "Spatially distributed tactile feedback for kinesthetic motion guidance," in *2010 IEEE Haptics Symposium*, pp. 519–526, 2010.

[4] L. Devigne, M. Aggravi, M. Bivaud, N. Balix, C. S. Teodorescu, T. Carlson, T. Spreters, C. Pacchierotti, and M. Babel, "Power wheelchair navigation assistance using wearable vibrotactile haptics," *IEEE transactions on haptics*, vol. 13, no. 1, pp. 52–58, 2020.

[5] J. J. Gil, I. Díaz, P. Ciáurriz, and M. Echeverría, "New driving control system with haptic feedback: Design and preliminary validation tests," *Transportation Research Part C: Emerging Technologies*, vol. 33, pp. 22–36, 2013.

[6] A. R. Krueger, P. Giannoni, V. Shah, M. Casadio, and R. A. Scheidt, "Supplemental vibrotactile feedback control of stabilization and reaching actions of the arm using limb state and position error encodings," *Journal of neuroengineering and rehabilitation*, vol. 14, pp. 1–23, 2017.

[7] E. Collaço, E. Kira, L. H. Sallaberry, A. C. Queiroz, M. A. Machado, O. Crivello Jr, and R. Tori, "Immersion and haptic feedback impacts on dental anesthesia technical skills virtual reality training," *Journal of Dental Education*, vol. 85, no. 4, pp. 589–598, 2021.

[8] J. H. Bultitude, G. Juravle, and C. Spence, "Tactile gap detection deteriorates during bimanual symmetrical movements under mirror visual feedback," *PloS one*, vol. 11, no. 1, p. e0146077, 2016.

[9] C. E. Chapman and E. Beauchamp, "Differential controls over tactile detection in humans by motor commands and peripheral reafference," *Journal of Neurophysiology*, vol. 96, no. 3, pp. 1664–1675, 2006.

[10] K. Bark, E. Hyman, F. Tan, E. Cha, S. A. Jax, L. J. Buxbaum, and K. J. Kuchenbecker, "Effects of vibrotactile feedback on human learning of arm motions," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 1, pp. 51–63, 2015.

[11] G. Juravle and C. Spence, "Juggling reveals a decisional component to tactile suppression," *Experimental Brain Research*, vol. 213, pp. 87–97, 2011.

[12] Z. A. Zook, J. J. Fleck, T. W. Tjandra, and M. K. O'Malley, "Effect of interference on multi-sensory haptic perception of stretch and squeeze," in *2019 IEEE World Haptics Conference (WHC)*, pp. 371–376, 2019.

[13] I. Oakley and J. Park, "Did you feel something? distracter tasks and the recognition of vibrotactile cues," *Interacting with Computers*, vol. 20, pp. 354–363, 2008.

[14] Q. Chen, S. T. Perrault, Q. Roy, and L. Wyse, "Effect of temporality, physical activity and cognitive load on spatiotemporal vibrotactile pattern recognition," in *Proceedings of the 2018 International Conference on Advanced Visual Interfaces*, AVI '18, (New York, NY, USA), Association for Computing Machinery, 2018.

[15] H. Z. Tan, C. M. Reed, and N. I. Durlach, "Optimum information transfer rates for communication through haptic and other sensory modalities," *IEEE Transactions on Haptics*, vol. 3, no. 2, pp. 98–108, 2009.

[16] K. Bark, P. Khanna, R. Irwin, P. Kapur, S. A. Jax, L. J. Buxbaum, and K. J. Kuchenbecker, "Lessons in using vibrotactile feedback to guide fast arm motions," in *2011 IEEE World Haptics Conference*, pp. 355–360, IEEE, 2011.

[17] J. V. Salazar Luces, K. Okabe, Y. Murao, and Y. Hirata, "A phantom-sensation based paradigm for continuous vibrotactile wrist guidance in two-dimensional space," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 163–170, 2018.

[18] H. Z. Tan, S. Choi, F. W. Lau, and F. Abnousi, "Methodology for maximizing information transmission of haptic devices: A survey," *Proceedings of the IEEE*, vol. 108, no. 6, pp. 945–965, 2020.

[19] G. Luzhnica, S. Stein, E. Veas, V. Pammer, J. Williamson, and R. M. Smith, "Personalising vibrotactile displays through perceptual sensitivity adjustment," in *Proceedings of the 2017 ACM International Symposium on Wearable Computers*, pp. 66–73, 2017.

[20] Z. Logan, Q. Deitrick, and K. Fitzsimons, "Evaluating factors affecting the perception of multi-sensory vibration and skin-squeeze cues during voluntary movement," *IEEE Transactions on Haptics*, pp. 1–8, 2025.

[21] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, vol. 8, no. 3, pp. 229–256, 1992.

[22] V. Konda and J. Tsitsiklis, "Actor-critic algorithms," *Advances in neural information processing systems*, vol. 12, 1999.

[23] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*, pp. 1928–1937, PmLR, 2016.

[24] R. S. Sutton, A. G. Barto, *et al.*, *Reinforcement learning: An introduction*, vol. 1. MIT press Cambridge, 1998.

[25] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proceedings of the 19th international conference on World wide web*, pp. 661–670, 2010.

[26] S. L. Scott, "A modern bayesian look at the multi-armed bandit," *Applied Stochastic Models in Business and Industry*, vol. 26, no. 6, pp. 639–658, 2010.

[27] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *International conference on machine learning*, pp. 1889–1897, PMLR, 2015.

[28] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[29] A. Israr and I. Poupyrev, "Tactile brush: Drawing on skin with a tactile grid display," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, (New York, NY, USA), p. 2019–2028, Association for Computing Machinery, 2011.

[30] J. Achiam, "Spinning Up in Deep Reinforcement Learning," *GitHub repository*, 2018.

[31] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021.

[32] P. Dhariwal, C. Hesse, O. Klimov, A. Nichol, M. Plappert, A. Radford, J. Schulman, S. Sidor, Y. Wu, and P. Zhokhov, "Openai baselines." `https://github.com/openai/baselines`, 2017.

[33] Z. A. Zook, O. O. Ozor-Ilo, G. T. Zook, and M. K. O'Malley, "Snaptics: Low-cost open-source hardware for wearable multi-sensory haptics," in *2021 IEEE World Haptics Conference (WHC)*, pp. 925–930, 2021.

[34] J. V. S. Luces, K. Okabe, Y. Murao, and Y. Hirata, "A phantom-sensation based paradigm for continuous vibrotactile wrist guidance in two-dimensional space," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 163–170, 2017.

[35] S. F. C. Gutierrez, J. V. S. Luces, and Y. Hirata, "Modality influence on the motor learning of ballroom dance with a mixed-reality human-machine interface," in *2022 IEEE/SICE International Symposium on System Integration (SII)*, pp. 177–182, 2022.

[36] Z. Liao, J. V. S. Luces, and Y. Hirata, "Human navigation using phantom tactile sensation based vibrotactile feedback," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5732–5739, 2020.

[37] J. Brooke *et al.*, "Sus-a quick and dirty usability scale," *Usability evaluation in industry*, vol. 189, no. 194, pp. 4–7, 1996.

[38] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information theory*, vol. 37, no. 1, pp. 145–151, 1991.

[39] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: A practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 57, pp. 289–300, 01 1995.

[40] J. Salazar, Y. Hirata, and K. Kosuge, "Motion guidance using haptic feedback based on vibrotactile illusions," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4685–4691, IEEE, 2016.