

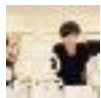
Reading 6

Discussion on Learning from Demonstration

[Subscribed](#)

- [Brief comments](#)

Display mode
newest first
form



Display replies flat, with oldest first
Display replies in threaded form

Display replies flat, with
Display replies in nested

Discussion on Learning from Demonstration
by [Bo Cao](#) - Friday, February 24, 2017, 9:34 AM

In "[A survey of robot learning from demonstration](#)." In general I think the idea of robot learning from demonstration is great, especially for reducing the training for robots as mentioned in Goodrich & Schultz: [Human-Robot Interaction: A Survey](#) (2007), which is one of the five attributes that a designer can affect.

"[A survey of robot learning from demonstration](#)." mentioned that the design choice also depends on programmer's preference, in my opinion, the end result of learning policy would reflect user's preference implicitly. In addition, from the talk by [Brad Hayes](#) on Thursday (02/23/2017), one question came to my mind was that could the policy for one user's be scaled to different robots even if the kinematics of these robots are different? For instance, I might used to collaborate with Baxter for assembling, and it had learned from me and developed a policy within the state context and actions, which was fit for me. But later it was broken for some reason and replaced with a KINOVA arm, the kinematics between these two are different, therefore sharing the policy from Baxter to KINOVA cannot be merely be a deep copy. In the other words, the *Recorded Mapping* is not exactly the same as the *Embodiment Mapping* as $g_R(z,a) \neq g_E(z,a)$. This might involve figuring out the level of the abstraction of behaviors since too much details of the movements of various joints from different robots might not be very interested to users as long as the abstract interaction is still what user likes. That is to say, we should figure out this level of abstraction of behaviors in the policy, so that $g_R(z,a) = g_E(z,a)$ in Imitation.

I think this survey is very comprehensive, but for me still it lacks a depth thought of dealing with uncertainty. In both 5.1 Undemonstrated State and 6.3 Execution Failures, one possible approach would be the learning from failure as explored in "[Robot learning from failed demonstrations](#)". Another approach could be to express explicitly their intentions of requiring demonstration from

teacher, and express explicitly their execution failures, such that the policy can be updated in this context.

In terms of "[The march of the robot dogs](#).", robot pet could provide companionship for people like the old, it should not replace this role. one direction designer could think for is to help socially isolated people interact with each other, by making them connected to each other, like the idea of POKEMON GO encouraging people exploring the real world together. Essentially, the emotion, the companionship should be provided with real person. As such, robot pet should be designed as a means, not an end of this purpose.