

Proposal

einer Master-Thesis mit dem Titel „Learning from Demonstration with Dynamic Trajectories“

B.Eng. Benjamin Reiner

13. August 2013

1 Problemstellung

Um einem modernen Roboter eine Tätigkeit beizubringen ist ein klassischer Programmieransatz oft zu umständlich zudem kann er nicht von einem Laien ausgeführt werden. Learning from Demonstration (LfD) ist eine praktikable Lösung für dieses Problem. Die Learning from Demonstration Methode Averaging Trajectories funktioniert in der bisherigen Form schon sehr gut. Mit diesem Verfahren können die Szenarien „Dose in Becher werfen“ und „Dose in Becher einschenken“ mit drei Demonstrationen gelernt werden. In meiner Projektarbeit „Robot independent ROS package for Learning from Demonstration“[4] habe ich die Arbeit von Heiko Posenauer[3], die sich mit dieser Methode befasst, auf ROS portiert und den Code Roboter unabhängig gestaltet. Der von Posenauer beschriebene Algorithmus beruht auf der Bedingung, dass die zu manipulierenden Objekte nicht verschoben werden, d.h. hat der Roboter die Objektpositionen erfasst und eine Person verschiebt aus versehen ein Objekt, verfehlt der Roboter das Objekt. Für die Alltagstauglichkeit des Verfahrens ist diese Bedingung nicht optimal.

2 Relevanz

Dieses Problem ist für das IKI¹ sehr interessant, da für das AsRoBe²-Projekt ein möglichst alltagstaugliches LfD-Verfahren nötig ist. Die Bedingung, dass die Objekte sich nicht bewegen dürfen verhindert auch die Interaktion mit Personen, da diese sich, nachdem die Objektpositionen erfasst wurden, nicht mehr bewegen dürfen. Eine Lösung dieses Problems würde es erlauben diese Mensch-Roboter-Interaktion zu evaluieren.

Mit der Lösung dieses Problems könnte gezeigt werden, dass ein einfaches Verfahren, wie das von Posenauer beschriebene, ähnlich gute Eigenschaften wie die komplexen Dynamic Movement Primitives (DMPs) besitzen kann.

¹Institut für Künstliche Intelligenz

²Assistenzroboter für Menschen mit körperlicher Behinderung

3 Stand nach der Arbeit

Mit Fertigstellung der hier vorgeschlagenen Arbeit soll es möglich sein bei der Reproduktion, einer mittels Learning from Demonstration by Averaging Trajectories (LfDaT) gelernten Trajektorie, die Objekte zu verschieben. Der Roboter erfasst die veränderten Objektpositionen selbstständig und passt die Trajektorie entsprechend an. Die Ergebnisse sollen mit einer ausführlichen empirischen Studie untersucht worden sein.

4 Ansatz

Um die veränderten Objektpositionen zu erfassen wird ein bestehendes Objekt-Tracking-Package verwendet. Mögliche Pakete hierfür wären `ar_pose`³ oder `ar_track_alvar`⁴. Hat sich mindestens eine Objektposition um mehr als einen gewissen Schwellwert verändert, wird der LfDaT-Algorithmus mit den neuen Positionsdaten erneut ausgeführt. Dieser Algorithmus soll dabei so performant sein, dass man ihn auch sehr schnell hintereinander ausführen kann. Wurde die Trajektorie bereits gestartet, so soll diese möglichst glatt an die neu berechnete Trajektorie angeglichen werden. Zu beachten ist hierbei, dass etwaige Constraints die durch die Objekte vorgegeben werden stark eingehalten werden. Das bedeutet, dass wenn der Greifer sich nahe an einem Objekt befindet, dass von der aktuellen Position aus, auf kürzestem Wege, in die neu berechnete Trajektorie gewechselt wird.

5 Vergleichbare Arbeiten

Wie bereits oben beschrieben basiert diese Arbeit auf der Bachelor-Thesis von Heiko Posenauer[3]. Diese Arbeit besitzt aber den in Kapitel 1 beschriebenen Nachteil. Posenauers Arbeit wiederum, baut auf den Ergebnissen der Master-Thesis von Markus Schneider[5] auf. Schneider verwendet in seiner Arbeit Gaußsche Prozesse.

Karl Glatz beschreibt in seiner Arbeit „Adaptive Learning from Demonstration using Dynamic Movement Primitives“[1] ein LfD-Verfahren, das es ebenfalls erlaubt Objekte zu verschieben. Sein Verfahren unterscheidet sich vom theoretischen Hintergrund her aber sehr von der hier vorgeschlagenen Arbeit. Glatz nutzt in seiner Arbeit DMPs, die mittels Locally Weighted Regression (LWR) gelernt werden. Dieses Verfahren ist mathematisch um einiges komplexer als das von Posenauer beschriebene, das mit der sehr einfachen Gaußschen-Normalverteilung arbeitet.

Einen ähnlichen Ansatz wie Glatz beschreiben Pastor et al. in ihrer Arbeit „Learning and Generalization of Motor Skills by Learning from Demonstration“[2]. Dort werden eigens angepasste DMPs mittels Linearer Regression gelernt. Bei diesem Verfahren ist es ebenfalls möglich die Objektpositionen während der Ausführung zu ändern. Außerdem werden störende Objekte im Bereich der Trajektorie berücksichtigt, sodass der Roboter-

³http://www.ros.org/wiki/ar_pose

⁴http://www.ros.org/wiki/ar_track_alvar

arm diesen ggf. ausweichen kann. Das Ausweichen funktioniert auch, wenn diese Objekte plötzlich auftauchen. (Video: <http://youtu.be/LuF1WNlcdfM>)

6 Meilensteine

1. Bestehende Codebasis auf Vordermann gebracht (KW 31)
 - a) Anpassungen an Katana Package in offiziellem Repo
 - b) Möglichkeit um Katana Trajektorie zu unterbrechen implementiert
 - c) LfD Code für Ausführung in Schleife vorbereitet
 - d) auf Groovy portiert
 - e) funktionsfähige Simulation vorhanden
2. Object Tracking einsatzbereit (KW 32)
 - a) Tracking packages verglichen
 - b) optimales Tracking package ausgewählt
 - c) gewähltes Tracking package lauffähig
3. Prototyp 1 fertig (KW 37) (Urlaub in KW 36)
 - a) Objektpositionen werden von Tracking-Software erhalten
 - b) Wenn sich ein Objekt verschiebt, wird die neu berechnete Trajektorie ausgegeben
 - c) Erkennt wenn ein Objekt gegriffen wurde
 - d) Bei gegriffenem Objekt wird letzte Position vor dem Greifen verwendet
4. Prototyp 2 fertig (KW 39)
 - a) Von aktueller Position aus wird auf neue Trajektorie übergegangen
 - b) Constraints werden sauber eingehalten
 - c) Ausführung auf Roboter möglich
5. Finaler Stand fertig (KW 41)
 - a) System besitzt alle geforderten Eigenschaften
 - b) Alle bekannten und relevanten Bugs behoben
 - c) Evaluation der Implementierung und der Verfahrens möglich
6. Verfahren Evaluiert (KW 44)
 - a) Anfälligkeit für Fehlverhalten evaluiert
 - b) Evaluiert wie weit bzw. schnell sich die Objekte bewegen dürfen

- c) Interaktion mit Menschen evaluiert
- 7. Gliederung der Thesis fertiggestellt (KW 44)
 - a) Mit Betreuer abgesprochen
- 8. Thesis ausformuliert (KW 50)
- 9. Thesis abgabebereit (KW 51)

Literatur

- [1] Karl Glatz. Adaptive learning from demonstration using dynamic movement primitives, August 2012.
- [2] Peter Pastor, H. Hoffmann, T. Asfour, and S. Schaal. Learning and generalization of motor skills by learning from demonstration. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 763–768, 2009.
- [3] Heiko Posenauer. Robot learning from demonstration by averaging trajectories, September 2012.
- [4] Benjamin Reiner. Robot independent ros package for learning from demonstration, March 2013.
- [5] Markus Schneider. Learning from demonstration with gaussian processes, September 2009.