How to embody AI in robotics?
On AI method engineering in robot engineering and human-robot interaction

Teade Punter[1], **Eric Dortmans**[1,2], **Sjriek Alers**[2], **Marijke Bergman**[3]
Fontys University of Applied Sciences
1. Lectoraat High Tech Embedded Software
2. Lectoraat Mechatronica en Robotica
3. Lectoraat Mens en Psychologie

{teade.punter, e.dortmans, s.alers, m.bergman}@fontys.nl

Keywords: autonomous robot, AI, machine learning, method engineering, human robot interaction, teaching.

1. Introduction

Robots don't fail! This is a common opinion in our community. However, robots must be programmed. For this the model that defines what the robot must do is of utmost importance. A model assumes what will happen if the robot has to perform its actions. However, the model is often not fully compliant, or is limited by what we know in advance about the situations the robot will encounter. The model will therefore be incomplete or difficult to define. This prevents an operating robot from getting into all kinds of undefined situations. It is therefore attractive to apply learning in robotics, or in other words, to apply AI in robot engineering.
Artificial Intelligence (AI) is the intelligence demonstrated by artefacts and algorithms, as opposed to humans and other living beings (Russell and Norvig, 2016). Most of the AI we hear about these days is designed to support online business processes, for example, chat bots to answer human questions or online shopping assistance and product recommendations. We focus on the embodiment of AI in robots (Hoffmann and Pfeifer, 2012). Robot AI is the ability of the robot to learn about its environment and act accordingly. It is embodied intelligence, because a robot is a cyber physical system. There are similarities to game AI, but robot AI works in the real world.

2. Impact of AI on engineering disciplines and human machine interaction

AI (will) impact(s) current engineering disciplines for robot development/engineering and the human machine interaction. Especially software engineering and mechanical engineering are impacted. When looking at software engineering we perceive that the value of AI is in its ability to create models to conduct actions and perform tasks without using explicit instructions. The decision taking mechanisms supervising an cyber physical system, like in IoT and robotics, is therefore impacted. Because AI relies on patterns and inference mechanisms. It is this ability that might lead to automating the software development. In a traditional software development project the architect and developer discover product features and requirements and programming them in a method. AI will change software development because of intelligence potential, see e.g., (Heck, 2019). The term "Software 2.0" coins to this development[1]. The impact of AI on Software engineers is that they will need to implement smart sensory data interpretation, modelling, decision making, behavior implementation and networking.
When looking at mechanical engineering control is key. Control system engineering models electro-mechanical systems, often by constructing formulas and models at a level of differential equation level. AI might help to learn these formulas and models from observing the system and the system parts, such as dedicated hardware or sensors. AI is an alternative for complex model building experience.
The impact of AI on Mechatronics engineers is that they will need to implement smart electro/mechanical robot bodies, smart sensory feedback and smart grasping and (loco)motion control.

[1] The prospect for software development should be seen in the context of the drive in software engineering to go for abstraction (Colburn and Shute, 2007).

Although the application of AI in software– and control engineering is not fully explored yet, its applicability seems to be infinite. Adding "intelligence" to machine will help to make them act autonomously, to diagnose problems themselves, as well as to make them self-repairing. The other way around, dealing with their world, robot are fed by environment data. When these data are gathered via machine learning / AI this will mean that robot development will face development hurdles. Robots have to deal with an unstructured world. We focus on embodied AI.

3. Embodied AI

Co-bots have to coexist and cooperate with humans. The users of co-bots often are less technically literate than the operators of traditional industrial robots. It is therefore of the utmost importance that humans can interact naturally and intuitively with co-bots. Furthermore they must fit in the human work environment. Though the larger body of research on Human Robot Interaction (HRI) and robot design focuses on service robots interacting in social or public settings, for instance the work of Dautenhahn (2007) and Sisbot et al (2010), the main findings will be valid for industrial co-bots as well.
Given that co-bots will work closely together with us, it may be assumed that several principles of human teamwork will apply in this situation as well. In this respect the concepts of interdependency, trust, communication and coordination are considered to be important.

In the co-bot architecture we can identify several places where AI can be applied, namely:

- Body control: health/vitals, safety and motion control.
- Perception: interpretation of sensor data, including localization, object detection, human intent recognition.
- World Model: making a model of the state of the world based on own perceptions and those of companion robots.
- Actuation: basic behavioral skills (including motion planning, collision avoidance, grasping) and behavior selection (which behavioral skill to activate depending on the state of the world).

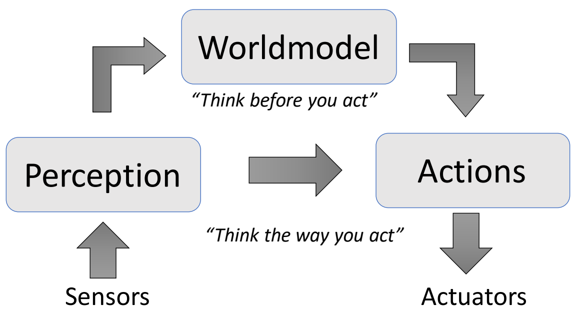

Figure 1 **Two application areas for AI in robotics: Perception AI and action AI.**

Meanwhile academic research has provided us with a large number of AI algorithms that can be applied in a co-bot setting. For example, Behavior Trees (Colledanchise & Ögren, 2018), Simultaneous Localisation And Mapping: SLAM (Thrun et al. 2010) and Path & motion Planning (LaValle, 2006). With the algorithms we can implement robot applications like the motion planning and collision avoidance of robot arm.

Applied AI research[2] as opposed to academic AI research is not focusing on the research for better algorithms. As applied AI researchers we focus on the practical aspects of applying AI algorithms:

- How to map a concrete co-bot use case on a set of problems that we need to solve using AI?
- Which of the known AI algorithms are the best choice to solve the problems at hand?

[2] In the Netherlands done by research groups in hoger beroepsonderwijs (hbo).

- What are the trade-offs between traditional algorithms and AI algorithms?
- How to combine traditional and AI algorithms?
- How do we get the training dataset needed for AI algorithms?
- How can we improve the quality and quantity of the data using cleaning and augmentation?
- How to setup an appropriate AI algorithm training and testing pipeline?
- How to validate a solution in terms of required quality, performance and reliability?

These questions relate to three of the four research challenges as identified by the AI Research Agenda for the Netherlands (NWO, 2019). The three facets that should be on the applied research agenda of the hbo as well are: Creating AI systems, AI systems and humans and AI systems and society. We consider the fourth challenge of the agenda "Creating AI Components", like new AI concepts and algorithms, to be the territory of fundamental academic research.

Concerning the research challenges on "Creating AI systems" we propose to work on an AI method engineering approach with other (applied) research and educational institutes. We argue that tailoring of AI methods and tools is needed to enable applicable, responsible and understandable AI in robotics. Our primary focus is to equip companies and our students with practical and validated AI knowledge, the methods and tools to apply AI.

4. AI Method engineering for robot engineering

Robot development and engineering should be a reasoning mechanism that consists of a framework to select existing AI methods and tools, like Keras, Pytorch or models and algorithms, for specific robot contexts. The approach should help companies and students to work with the new terminology and to find their way in the many different AI methods can be used for robot design. Each method might cover its own problem area, while repeated application will provide more validated and appropriate application. Selecting and especially combining (parts of) methods is one of the educational challenges when designing and engineering systems.

Two typical examples of AI application in robotics that were conducted by our research groups and that might be input for AI method engineering are:

- Localization in robot soccer (Robocup MSL). A setup was made by using computer vision and applying machine learning on the camera images. The information is fused with data from the odometry and gyro of the robot to achieve more accurate positioning than can be done with existing sensors.
- Interaction Module for Automated Guided Vehicles. A setup was developed by using an neural network for pose recognition. The network was implemented via an Interaction Module that is mounted on a mobile robot to detect human intentions, e.g., if a person is approaching or leaving the surroundings of the robot.

The AI method engineering approach distinguishes the application domain and the framework. The application domain is the (problem)area in which AI is used for improving a robot domain. An important aspect of the AI applications is that it is context-bound, so what works "well" depends on the situation. Therefore we are interested in when AI method parts work and when they will not work.
The framework for AI application is the thinking about how to design and apply the AI methods and tools. It can is method engineering to select appropriate AI methods and tools, to apply and to learn about them. An overview of this version of the approach is depicted below[3].
Phases in the application domain that will be most like are:

[3] This is a version based on current thoughts. The foreseen research might results in changes of the approach.

- Domain analysis – analysis of the robot challenge. What to achieve with the robot and which strategy should be followed for an AI implementation to support the robot or human–robot interaction?
- Data preparation – preparation, collection, cleaning and processing of data. Preparation - A machine learning model needs to be trained using the data it will also use for the predictions.
- Design, deployment – the design of the model.
- Feedback on the AI implementation.

Phases in the Framework for AI application that will be most like are:
- Selection of AI method – selection of part of existing tools, framework to implement the strategy chosen for the domain analysis.
- Experiences with AI method/tool – based on the experiences gathered during the design / deployment as well as on the implementation in the field (feedback).
- AI method evaluation – based on the experiences. Lessons learning / guidelines for the toolbox.

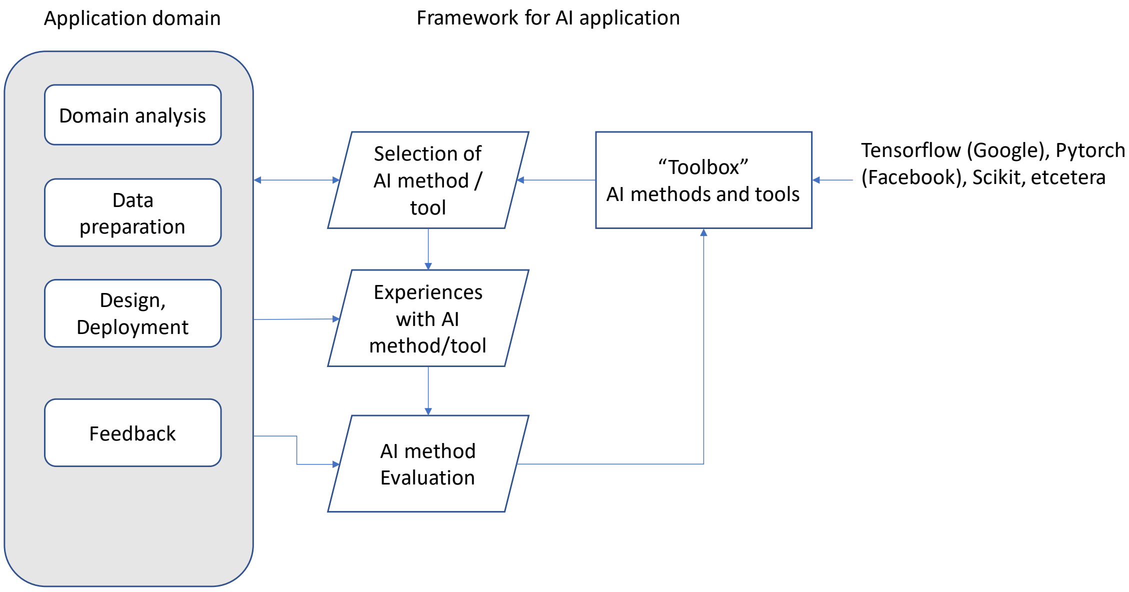

Figure 2 Overview of AI Method engineering for robot engineering.

5. Summary
We think that a structured approach is needed for embodying AI in robotics. This approach should help the companies we work with, which are often no AI-experts, and should facilitate our students. Therefore we argue that tailoring of AI methods and tools is needed to enable applicable, explainable AI in robotics. The approach should be applicable by companies and students to address practical problems at hand. The approach should be explainable because its models should be interpretable and inclusive and should be deployed with confidence[4]. Because embodiment requires insight in usefulness of robots and AI in their specific contexts, the approach requires the input of multiple research and educational areas, like mechatronics, psychology and software engineering. This is why we invite applied researchers to work on AI robot engineering and elaborate with us.

[4] https://cloud.google.com/explainable-ai/

References

- Colburn, T. G. Shute. 2007. Abstraction in Computer Science. *Minds & Machines, 17 (169)*. Available from: https://doi.org/10.1007/s11023-007-9061-7.
- Colledanchise, M., P. Ögren. 2018. *Behavior Trees in Robotics and Al: An Introduction*. Boca Raton, FL: CRC Press.
- Dautenhahn, K., 2007. Socially intelligent robots: dimensions of human–robot interaction. *Philosophical Transactions of the Royal Society: Biological Sciences*, 362(1480), pp.679-704.
- Dortmans, E. 2019. Robot AI – How to teach my robot? Available from: https://www.htesfontysict.nl/site/assets/files/1131/2019_dortmans_robotai_iip.pdf.
- Heck, P. 2019. *Software engineering for machine learning application*. Available from: https://fontysblogt.nl/software-engineering-for-machine-learning-applications/.
- Hoffmann M., R. Pfeifer. 2012. *The implications of embodiment for behavior and cognition: animal and robotic case studies*. Available from: https://arxiv.org/abs/1202.0440.
- LaValle, S. M. (2006). *Planning Algorithms*. Cambridge, England: Cambridge University Press.
- NWO. 2019. *Artificial Intelligence Research Agenda for the Netherlands* (AIREA-NL). Available from: https://www.nwo.nl/en/news-and-events/news/2019/11/first-national-research-agenda-for- artificial-intelligence.html
- Russell, S., P. Norvig. 2016. *Artificial intelligence: a modern approach*. Upper Saddle River: Pearson.
- Sisbot, E.A., L.F. Marin-Urias, X. Broquere, X., D. Sidobre, R. Alami. 2010. Synthesizing robot motions adapted to human presence. *International Journal of Social Robotics*, 2(3), pp.329-343.
- Thrun, S., W. Burgard, D. Fox. 2010. *Probabilistic robotics*. Cambridge, Mass.: MIT Press.

Acknowledgements

This work is conducted as work of the Fontys' Intelligent Robots project which was funded by the Fontys TEC for Society High Tech Systems and Materials programme.