MIN-Fakultät
Fachbereich Informatik
TAMS

Imitation Learning

Initial Concept and Approaches

Nguyen, Thi Linh Chi

Outline

- Motivation
- Basics and Definition
- Approaches & Examples
- Conclusion

Motivation

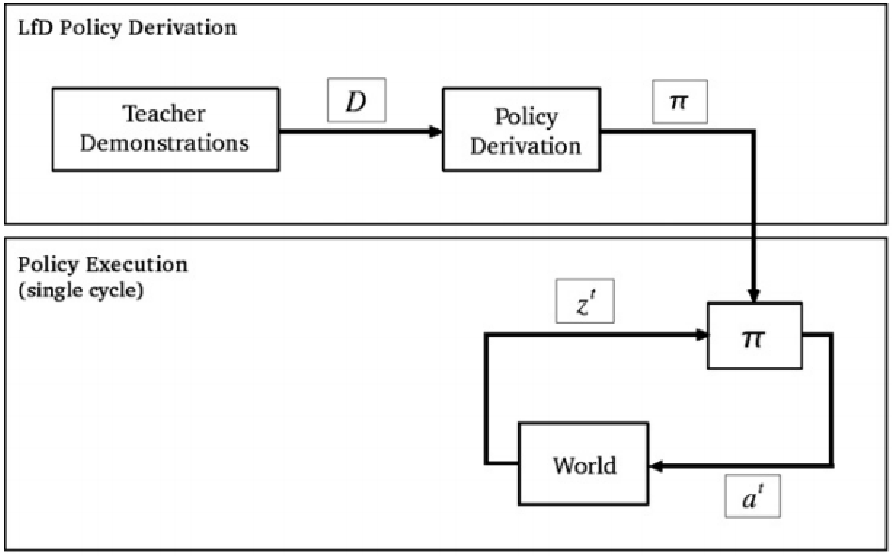
- Imitation Learning is a basic robotic learning method
- Not all animals can imitate
- Open door for non-robotic-experts to do research on robotics

Basics and Definition (1)

- "Imitation Learning is a means of learning and developing new skills from observing these skills performed by another agent." [2]
- Other terms: Learning from Demonstration, Learning by Observation, etc.
- Demonstration
  - Who involve?
  - What to demonstrate?
  - How to demonstrate?
    - Tele-operate
    - Shadowing

Basics and Definition (2)

- D: Demonstration
- z^t : observed state
- a^t : action
- π : policy



Control policy derivation and execution [1]

Approaches

- Three core approaches:
  - Mapping Function
  - System Model
  - Plans

Taxonomy

Approaches	Learning Techniques	
Mapping functions	Classification	Low Level Robot Actions
		Basic High Level Actions
		Complex High Level Action
	Regression (Mapping Functions Approximation)	At Run Time
		Prior Run Time
		Prior Execution Time
System Models	Reward Based Learning	Engineering Reward Functions
		Learning Reward Functions
Plans	Using Planner	

Mapping Function Approach (1)

- Directly map from state to action

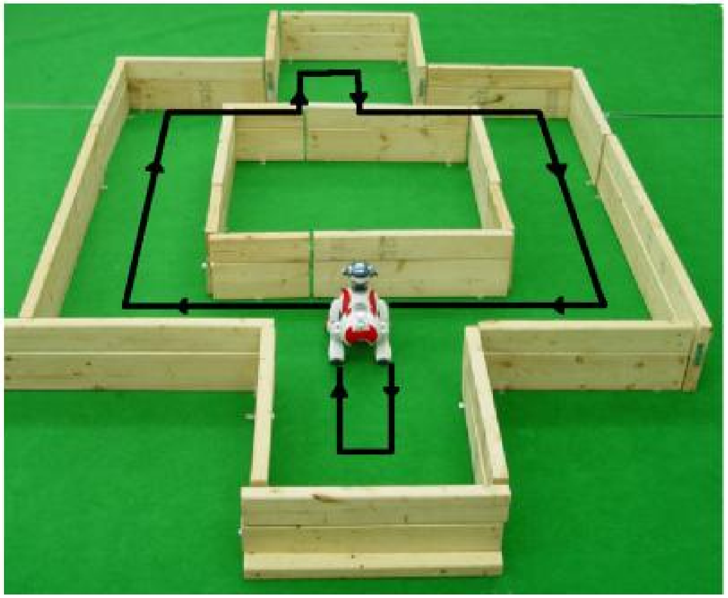
- 2 categories:
  - Classification
  - Regression

Mapping Function Approach (2)

	Classification	Regression
Input	Robot states Categorized input values	Robot states Non-categorized input values
Output	Robot actions Discreet value	Multiple demonstration set of Robot actions Continuous
Application	3 level of actions: - Low Level - Basic high level - Complex high level	Typically low level motions / behaviors - Imitate prior run time - Imitate at run time - Imitate prior execution time

Classification low level action example

- Low-level actions: basic commands such as moving forward or turning
- Corridor Navigation Domain [4]:



Classification high basic level action example

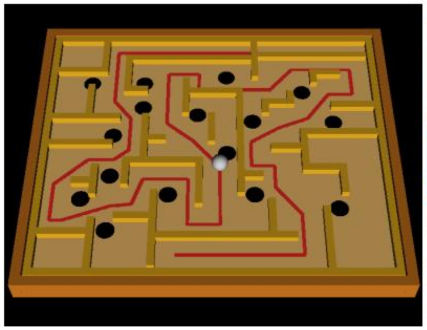
- Basic high level actions: motion primitives are composed or sequenced together
- Autonomous egg flipping [5]:

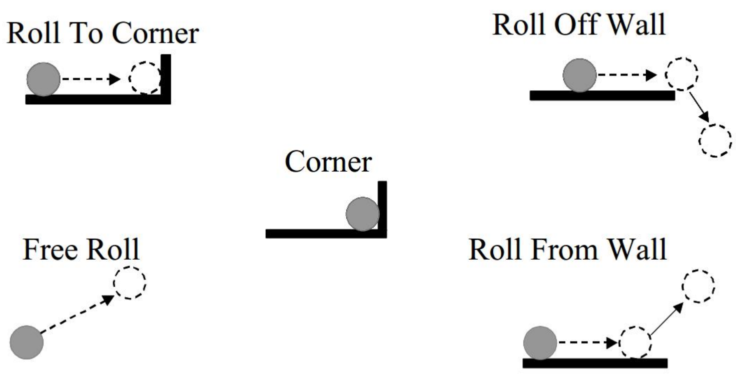
Classification complex high level action example

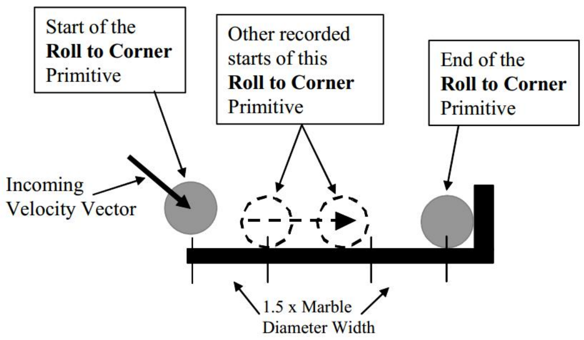
- Complex level control actions: behaviors are developed prior to task learning
- Robots co-ordination to sort balls [6]:

Regression at Run Time example

- Learning from demonstration through marble maze [7]:

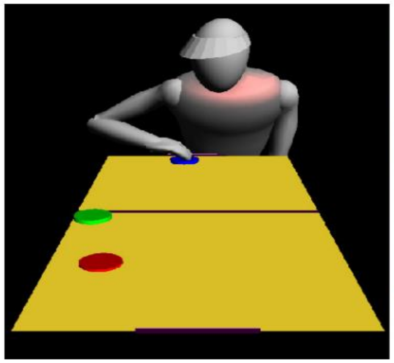


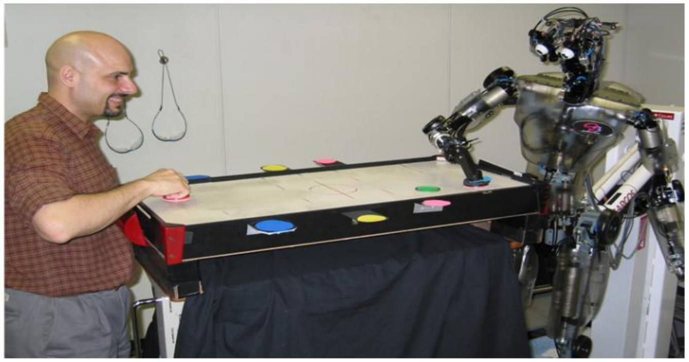


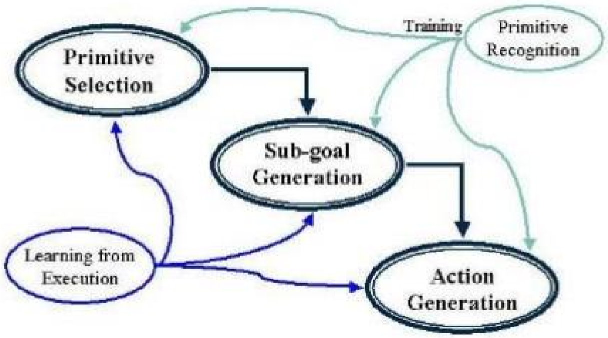


Regression prior Run Time example

- Humanoid plays air hockey [7]:

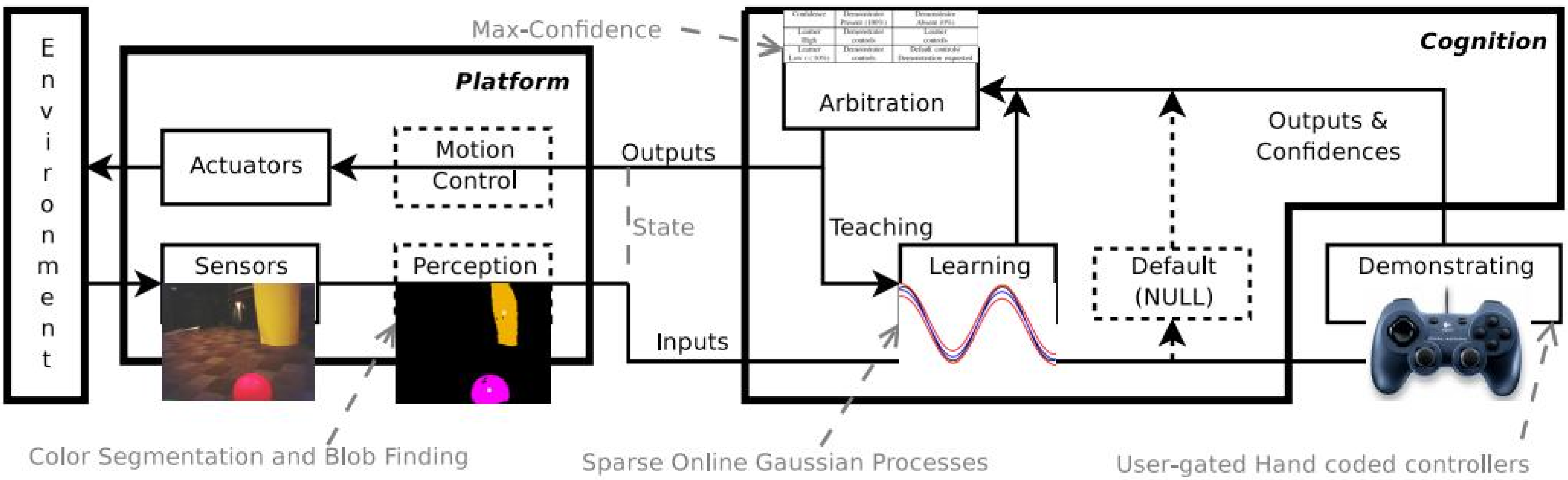


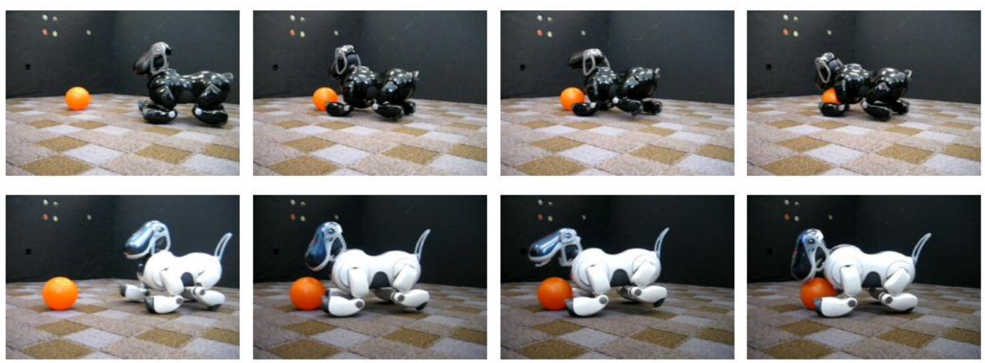




Regression prior Execution Time example

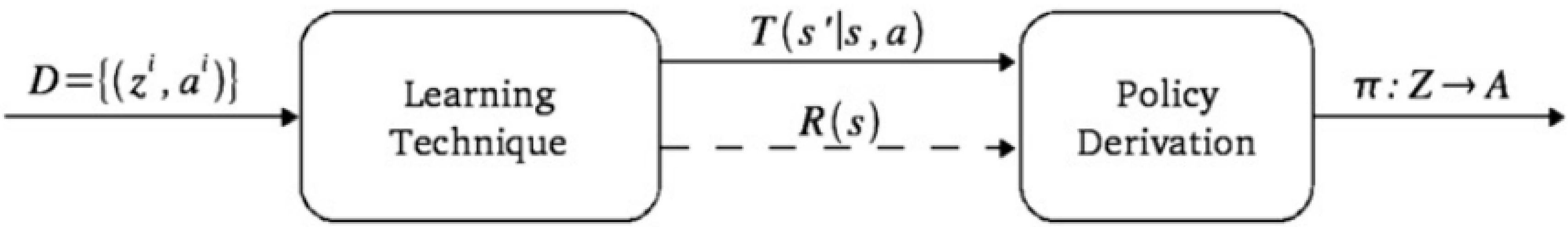
- Learning Robot Soccer Skills from Demonstration [12]:





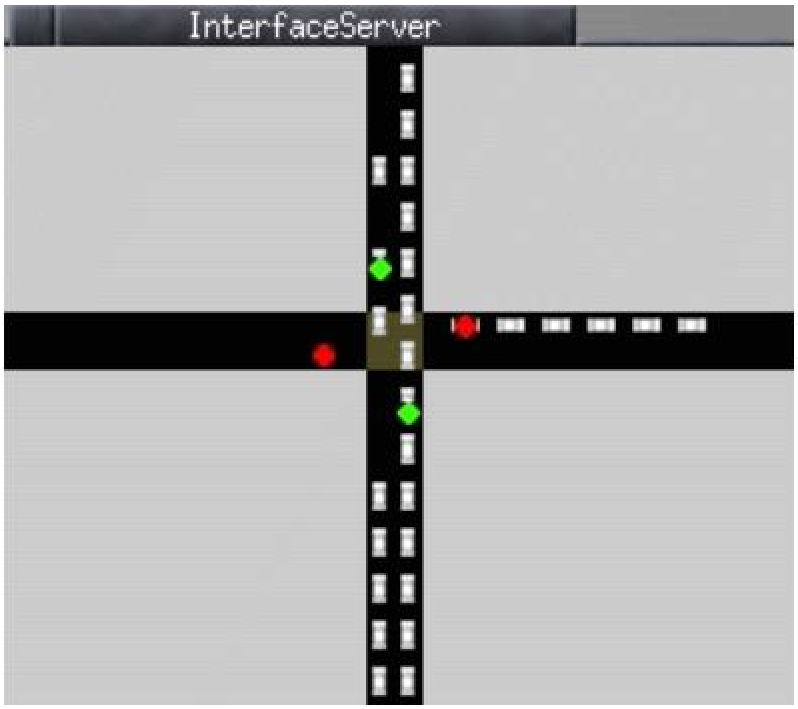
System Model Approach

- Imitate through a world dynamic model T and reward function R



System Model Approach Example

- Engineered reward functions: Traffic Simulator [8]



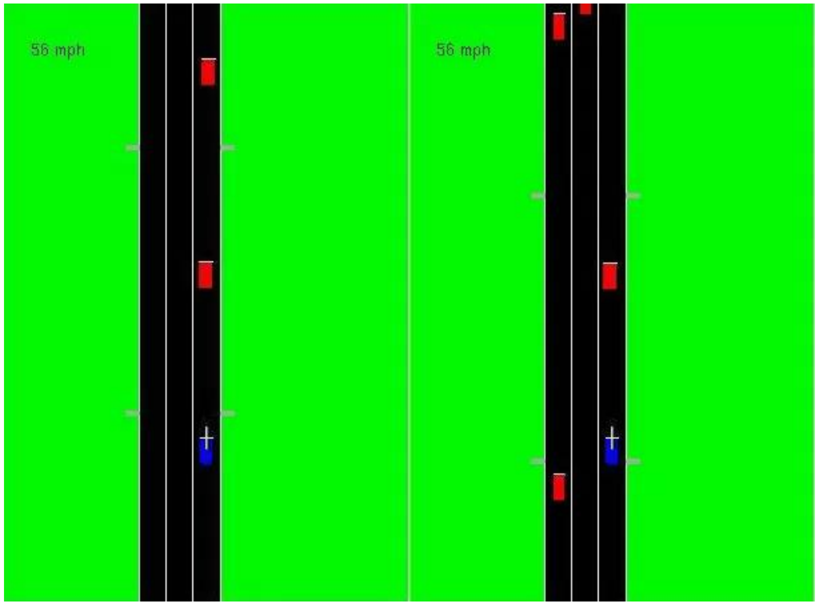
Graphic interface of Traffic Simulator

1. Agent i: receive the best average quality (bq_j) from all other agents ($j \neq i$). Quality for Agent i is cq_i.
2. Agent i: get state s for evaluation.
3. Agent i: calculate $k = \arg\max_j(bq_j)$, for all agents ($j \neq i$).
4. Agent i: if $cq_i < d\ max(bq_j)$:
    a. Agent i: send agent k the current state s and request advice.
    b. Agent k: switch to best parameters and run state s to produce its best guess at the adequate response (g).
    c. Agent k: return g to Agent i.
    d. Agent i: process advice (g).
5. Agent i: run state s and produce response g'.

Step of advice exchange between agents

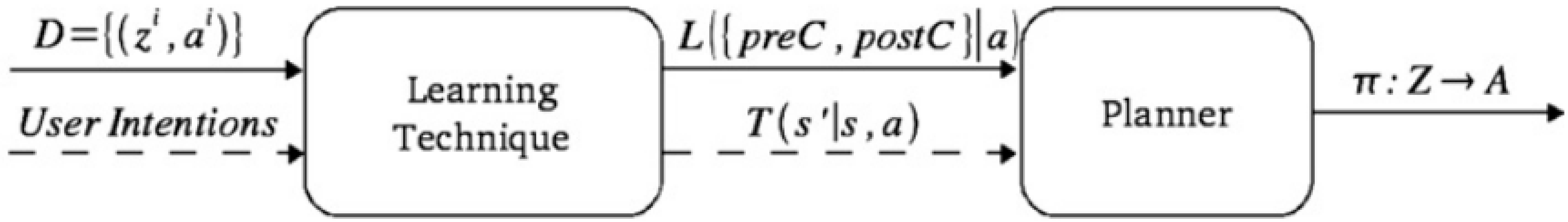
System Model Approach Example

- Learned reward functions: Car Driving Simulator [9]



Plans Approach

- Imitate through a state transition model T and set L of pre-conditions and post-conditions of action A



Plans Approach Example (1)

- Robot with ball collection task [10]

Algorithm 1 Precondition and Effect Filling

Input: Grounded actions $\langle a_1, \dots, a_N \rangle$ from demonstration
Input: Preceding and Succeeding states $\langle \{^-S_{a_1}, ^+S_{a_1}\} \dots \{^-S_{a_N}, ^+S_{a_N}\} \rangle$ for each action
Output: Grounded actions $\langle a_1, \dots, a_N \rangle$ with filled preconditions and effects

1: GROUP grounded actions into Operators $O_1, \dots, O_k$, s.t. $\forall O_{op, op=1 \dots k}, \nexists a_j, a_k \{a_j, a_k \in op\}$, SUBSTITUTE$(a_j, a_k)$ is invalid
2: **For all** Operators op **do**
3: Collect the action states $\{^-S_{a_j}, ^+S_{a_j}\} \forall_j a_j \in op$.
4: Remove inconsistent action states.
5: **For all** Operators op **do**
6: Get $Preconditions^{op} \longleftarrow {}^-S_{a_1} \wedge \dots \wedge {}^-S_{a_k}, a_1, \dots, a_k \in op$
7: Effects:
8: **If** exists effect $e^x_{a_j}, a_j \in op \wedge \exists a_k \in op$ where $\forall_y e^y_{a_k}$ SUBSTITUTE$(e^x_{a_j}, e^y_{a_k})$ is invalid **then**
9: **If** exists predicates $c^{W=\{w_1,\dots,w_m\}} \in {}^-S_{a_j}$ where $\arg(e^x_{a_j}) \cap \arg(c^W) \neq \emptyset \wedge \forall_{i, i \neq j}$, SUBSTITUTE$(e^x_{a_j}, e^y a_i)$ is invalid $\wedge\ c^W \ni {}^-S_{a_i}$, where $a_i, a_j \in op$ **then**
10: Add conditional Effect $condEffect^{op} \longleftarrow \{c^W, e^x_{a_j}\}$
11: **Else**
12: Add disjunctive Effect $Effects^{op} \longleftarrow e^x_{a_j} \vee Effects^{op}$
13: **Else If** $\exists$ effect $e^x_{a_j} \in \forall_i \{\triangle \langle ^-S_{a_i}, ^+S_{a_i} \rangle\} a_i, a_j \in op$ **then**
14: Add conjunctive Effect $Effects^{op} \longleftarrow e^x_{a_j} \wedge Effects^{op}$
15: Fill in Preconditions and Effects for each action $a_j \in op$

Plans Approach Example (2)

- Robot with ball collection task

Algorithm 2 Learning Looping Plans from Example

Input: Partial Order (PO) Graph
Output: Generalized Looping Plan
Transitively reduce PO Graph
Parameterize trace step actions
Detect LOOPS(Actions $a_1, \ldots, a_N$)
Order Steps by links.

Pick Object | Carry Object | Drop Object

Evaluation

- In common:
  - Advantages:
    - An easy learning method for robots
    - Rely on instructor experience and goodwill
  - Disadvantages:
    - Learning quality affected by teacher's performance
    - Hard to obtain correct demonstration if the task is complex
    - Things that cannot be learned through imitation
- Why does Imitation Learning open spaces for non-roboticists to participate?
- What is the best approaches?

Summary

- Introduced Imitation learning method
- Introduced approaches
- Examples in robotics

Literature

1. Argall, B. D., Chernova, S., Veloso, M., & Browning, B. (2009). A survey of robot learning from demonstration. *Robotics and autonomous systems*, *57*(5), 469-483.
2. Seel, N. M. (Ed.). (2012). *Encyclopedia of the Sciences of Learning*. Springer Science & Business Media.
3. Siciliano, B., & Khatib, O. (Eds.). (2008). *Springer handbook of robotics*. Springer Science & Business Media.
4. Chernova, S., & Veloso, M. (2007, May). Confidence-based policy learning from demonstration using gaussian mixture models. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*(p. 233). ACM.
5. Pook, P. K., & Ballard, D. H. (1993, May). Recognizing teleoperated manipulations. In Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on (pp. 578-585). IEEE.
6. Chernova, S., & Veloso, M. (2008, May). Teaching multi-robot coordination using demonstration of communication and state sharing. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 3* (pp. 1183-1186). International Foundation for Autonomous Agents and Multiagent Systems.
7. Bentivegna, D. C., & Atkeson, C. G. (2003, January). A framework for learning from observation using primitives. In *RoboCup 2002: Robot Soccer World Cup VI* (pp. 263-270). Springer Berlin Heidelberg.
8. Nunes, L., & Oliveira, E. (2004, July). Learning from multiple sources. In*Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 3* (pp. 1106-1113). IEEE Computer Society.
9. Abbeel, P., & Ng, A. Y. (2004, July). Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning* (p. 1). ACM..
10. Veeraraghavan, H., & Veloso, M. (2008, May). Teaching sequential tasks with repetition through demonstration. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 3* (pp. 1357-1360). International Foundation for Autonomous Agents and Multiagent Systems.

Literature

11. Grollman, D. H., & Jenkins, O. C. (2007, July). Learning robot soccer skills from demonstration. In *Development and Learning, 2007. ICDL 2007. IEEE 6th International Conference on* (pp. 276-281). IEEE.

The End

Thank you for your attention.

Any question?