Deep Learning for Complexity and Capability in Humanoid Robots

Powered by maxon motor

driven by precision
Welcome

- Overview of the particular type of robot we manufacture
  - An “Anthropomimetic” robot
- Place it within the range of humanoid robot technologies
  - Challenges and opportunities abound
- Make some predictions as to how robots may develop
  - The rise of smart robots
- Present our thoughts on embedded AI systems
- Q&A
Open Source Android

- Our robot is the humanoid with a single green eye
- Aims to copy the anatomy of the real human body
  - The “anthropomimetic” approach
- Developed a series of robot prototypes over the last 10 years
- Anatomical input from medical, sports training and plastinations
CRONOS – a conscious robot

- Original reason to create these robots was to investigate consciousness
  - Funded by the EPSRC Adventure Fund
  - Theory is to shape the sensation of existence with the form of the body
    - Make a robot as much like the human body as possible
    - Synthetic Methodology
- Embedded Intelligence
  - HAL is impossible
The Anthropomimetic Approach

- Endoskeleton actuated by tendons
  - Electric motors spool tendons
- 3D joints are true ball and socket
- Attachment points match the real muscles
  - Muscle does a lot more than just contract
- Compliant structure
  - Tendons are elastic
- Antagonistic set-up
Unexpectedly likeable robot

Despite looking like a skinned body, people loved the robot.
Control

- We were not constrained by having to actually make the robot do any particular task
  - The point of ECCE was to address “Emerging Cognition”
    - Had to build a structure that can be controlled
    - All complex animals solve this problem
- Doesn’t seem like a good idea to an engineer
  - Appropriate for copying the human body
    - “Not an engineered system!”
- Needs a self-learning control system
Robots with real capability

- Lots of robots have come and gone
  - End up abandoned in the corner
- World’s most popular robot is R2-D2
  - Not C3-PO – function is more important than form
- General Purpose Power Tool
  - From the POV of the user the task is completed in its entirety
  - Not an appliance
- Extremely high expectations – oldest anticipated invention
Technologies in humanoid robots

- A spectrum defined by how hard they are to control
  - Depends on mechanical design approach
- Classical
  - Familiar and well understood – factory robots
- Compliant
  - Series-elastic actuation – factory robots with suspension
- Bio-inspired
  - Compliant, tendon-driven, antagonistic
Classical robots – Common in industry

- Mechanical design maximises stiffness
- Very precise control
- Continuous maximum power
- Usually fixed base
- Highly dangerous
  - Contact with people heavily restricted by safety legislation
- Ultimate example ASIMO
  - Requires precise conditions to operate
ASIMO

- ASIMO Cart bot video
- Mobile phone catch
Compliant robots – Series Elastic

- Mechanical design maximises efficiency
- Precise control is difficult
- Peak power output considerably higher than average
- Well suited to legged robots
- Generally safer
  - Governed by difference legislation
- Best examples produced by Boston Dynamics
  - SPOT Mini is the most important
Compliant robots

- Spot video
- Flip video
Bio-inspired robots – ballistic tendons

- Mechanical design reproduces organic system
- Any control is difficult
- Ballistic speeds and power are possible
- Best design for legged robots
- Accidental high impacts rare but possible
  - Governed by compliant legislation
- Most complete examples ourselves and Kijiro
Ballistic robots

- Video ball throw
- Video jumping robot
Future direction

- Must be compliant
  - Real compliance, not active
- Tendons as appropriate
  - Hybrid design to save cost and complexity
- Tremendous amount of work to do in control
  - High level – vision, navigation, decision making, etc
  - Low level – , basic co-ordination, physics modelling, etc
- Are GPU’s the answer?
Complexity and capability

- By definition a general purpose robot cannot be simplified
  - Complexity at three levels: Mechanics, Electrical and Computational
- Dependency Principle
  - The interaction of three domains makes debugging very challenging
  - Lack of abstraction makes it hard to divide work
- Gestalt entity
  - Really don’t know what you have until it’s finished
  - When initial assumptions revisited you know you’re getting close
- Useful also means powerful enough to be dangerous
Practical solutions

- Convergence of culture and technology inevitable
- Two approaches evident:
  - 1. End-to-end development – Centralised resources e.g. Softbank
  - 2. Open Source movement – Distributed effort, worked for drones
- Parallels to iPhone and android, Mac and PC
  - Component level manufacturing remains black-box
- We’re looking for collaborators to move up to the next step
  - Enough reliable robots to generate the data for control
Problem

- We want the robots to perform tasks well in our environment, be safe and robust.
- Complexity of the robot: non-linearity, elasticity, compliance.
- Unpredictable environment, always changing.
- => not possible to program by hand
- The robot has to learn skills by itself
- Tools available for that: Reinforcement Learning
Applying RL on hardware directly

- Takes a **long time** to get results
- Requires **many robots** to learn faster
- In many cases, it would **damage itself or the environment**
- Needs of **maintenance and human supervision**
- Cost a lot of **money**
- Not efficient!
- We need to **simulate** the robot and apply RL in a simulation, then **transfer the skill** to the robot.
Procedure and progress

✓ Build the robot
✓ Getting actuators and sensors working
➢ Simulation of the robot
× Training of DRL in the cloud in a parallel simulators, on a server with GPUs
× Knowledge transfer to the real robot.
× Running inferences locally on a Jetson TX2
Jetson TX2 – a game changer

- Credit card size footprint
- 1TFLOPS with GPU
  - Run inferences in embedded systems
- 2 CAN bus
  - Direct connexion with motor controllers, no need of extra micro-controller hardware.
Technology convergence

- ROS
- BULLET Physics Library
- MuJoCo advanced physics simulation
- OpenSim
- PhysX by NVIDIA
- UE4 Unreal Engine
- maxon motor
Reinforcement Learning

- Blender + molecular add-on + python scripting + tensorflow
Deep Reinforcement Learning

- Combining DL and RL
- Example: The agent input observation from the environment is camera images on which DL with CNN can be applied.
Challenge

- Model the physics of series-elastic actuators
  - Blender + python scripting
  - Bullet API
  - OpenSim
- Simulators
  - ROS rviz – too limited in terms of physics
  - Gazebo – too limited in terms of graphics
- Model photo-realistic environment
  - Unreal Engine 4
- ISSAC Initiative
Thank you! Any questions?

- Contacts:
  - Rob Knight
    - rob.knight@therobotstudio.com
  - Cyril Jourdan
    - cyril.jourdan@therobotstudio.com

- Website: www.therobotstudio.com