Operating Humanoid Robots:
Comprehensive Modular Open Source Software for Humanoid Avatar Robots based on ROS

Stefan Kohlbrecher
Alberto Romay
Philipp Schillinger
Alexander Stumpf

Oskar von Stryk

www.sim.tu-darmstadt.de
Motivation

- Humanoid robots are suitable for **human tasks in human environments**:
  - Home
  - Industrial Environments
  - Disaster Response

Don Joven Agravante et al.
[https://youtu.be/-1BcC3aEuZM](https://youtu.be/-1BcC3aEuZM)
Human Environment Example

Driving Cars
Human Environment Example

Driving Cars
Challenges for Humanoid Robots
Challenges for Humanoid Robots

• Motions with multiple contacts (e.g. using handrails)

Versatile and robust (Loco-)Motion
Challenges for Humanoid Robots

- Motions with multiple contacts (e.g. using handrails)
- Ladders, uneven terrain and stairs
Challenges for Humanoid Robots

- Motions with multiple contacts (e.g. using handrails)
- Ladders, uneven terrain and stairs
- Doors

Versatile and robust (Loco-)Motion
Challenges for Humanoid Robots

Versatile and robust Perception

Versatile and robust (Loco-)Motion

- Environment for locomotion
- Objects for manipulation
- Ability to acquire new objects and their potential purposes on the fly
- Robustness to different lightning conditions
Challenges for Humanoid Robots

- Versatile and robust Perception
- Versatile and robust (Loco-)Motion
- Versatile and robust Manipulation

- Many different tools, only few exactly known in advance
- Acquiring new manipulation modes
- Ability to coordinate manipulation, locomotion & active perception
Challenges for Humanoid Robots

Versatile and robust Perception

Versatile and robust Manipulation

Versatile and robust (Loco-)Motion

Efficient Supervision via Human-Robot-Interaction

- Matching human and robot abilities best
- Appropriate levels of human-robot-interaction for highly diverse tasks
- Distribution between work tasks robot onboard and offboard (OCS)
Challenges for Humanoid Robots

- Versatile and robust Perception
- Versatile and robust Manipulation
- Versatile and robust (Loco-)Motion
- Efficient Supervision via Human-Robot-Interaction
- Limited Wireless Communication: bandwidth, latency, dropouts
Human Operator Perspective
DRC Finals (2015) Example
Humanoid Robots Requires Complex Software

- Re-Inventions are the time sink #1
- Progress requires...
  - Documentation (e.g. Papers)
  - Shared Software (e.g. Open Source Code)
  - Maintainers (e.g. the Community)
Notable Open Source Efforts Usable for Humanoid Robots

- **MIT:**
  - Pronto State Estimator
  - Drake Planning and Control
  - Director UI

- **IHMC:**
  - IHMC Controller
  - SCS Simulator

- **MoveIt!** – Manipulation planning

- **Gazebo** – Simulation including physics engines

- **ROS** – Robot Operating System ( Middleware)
Why ROS?

Prevent the Re-Invention of the Wheel!

- Common Ecosystem
  - Using common, well-defined interfaces
- Reusability of Software
System Architecture using ROS
Team ViGIR DRC Setup

OCS ROS network
Operator Control Station Computers
- Supervisor OCS
- Primary OCS
- Secondary OCS
- Gigabit Ethernet Switch

Onboard ROS Network
Field Computer(s)
- Perception
- Motion Control
- Planning
- Gigabit Ethernet Switch

Robot
- Main Controller
- Hands
- Multisense
- SA Cameras
- Gigabit Ethernet Switch

Traffic Shaper
Our Contributions (Overview)

**Versatile and robust Perception**
- Terrain Modeling

**Versatile and robust (Loco-)Motion**
- 3D Footstep Planning in rough terrain

**Versatile and robust Manipulation**
- Template-based Manipulation

**Efficient Human-Robot-Interaction**
- “Ghost Robot”
- Sliding Autonomy

Our Contributions (Overview)

Versatile and robust Perception
• Terrain Modeling

Versatile and robust (Loco-)Motion
• 3D Footstep Planning in rough terrain

Versatile and robust Manipulation
• Template-based Manipulation

Efficient Human-Robot-Interaction
• “Ghost Robot”
• Sliding Autonomy

Terrain Modeling

- Only point clouds required as input
- Uses Oct-Tree as data representation for efficient data lookup

https://github.com/team-vigir/vigir_terrain_classifier
Terrain Modeling
Online Generation

Surface Normal Estimation

Elevation Map

>> 4x
Our Contributions (Overview)

Versatile and robust Perception
- Terrain Modeling

Versatile and robust Manipulation
- Template-based Manipulation

Versatile and robust (Loco-)Motion
- 3D Footstep Planning in Rough Terrain

Efficient Human-Robot-Interaction
- “Ghost Robot”
- Sliding Autonomy

3D Footstep Planning

- Generates suitable sequence of full 3D (6 DoF) foot poses
  - Using A*-search-based planning approach
  - Novel collision check strategy allows for overhanging steps
- Adaptable to many bipedal robots

\[
s' = (x', y', \theta') \quad \quad s = (x, y, \theta) \quad \quad a = (\Delta x, \Delta y, \Delta \theta)
\]

Discrete Foot Placements


http://wiki.ros.org/vigir_footstep_planning
3D Footstep Planning

Example

- Robot's field of view
3D Footstep Planning

Example
3D Footstep Planning
Human Supervision

- Support for Interactive Footstep Planning
3D Footstep Planning
Available as Customizable Framework

- Modular and adaptable for any humanoid robot via plugins

---

Please visit Poster on Thursday 16:30-18:00 (ThPoS.23)

---

Our Contributions (Overview)

Versatile and robust Perception
- Terrain Modeling

Versatile and robust (Loco-)Motion
- 3D Footstep Planning in rough terrain

Versatile and robust Manipulation
- Template-based Manipulation

Efficient Human-Robot-Interaction
- “Ghost Robot”
- Sliding Autonomy

Template-Based Manipulation

- **Grasp template**
  - Potential grasp poses
  - Finger joint positions
  - Type of grasp
  - Potential stand poses

- **Stand template**
  - Potential robot poses

- **Object template**

https://github.com/team-vigir/vigir_object_template_manager
### Template-Based Manipulation

**Actions Over Object Templates**

<table>
<thead>
<tr>
<th>Action</th>
<th>Template</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Drill</td>
<td>$U_1 = {1, 0, 0, 0, 0, 0}$</td>
<td>The drill action possibility is a translation along the X axis (green arrow).</td>
</tr>
<tr>
<td>Door</td>
<td>$U_1 = {0, 0, 0, 1, 0}$, $U_2 = {0, 0, 0, 0, 1}$</td>
<td>The door action possibilities are to rotate around the Y axis (red ring) in $U_1$ and rotate around the Z axis (blue ring) in $U_2$.</td>
</tr>
<tr>
<td>Hose</td>
<td>$U_1 = {1, 0, 0, 1, 0}$</td>
<td>The hose action possibility is a translation and a rotation around the X axis (green arrow and ring).</td>
</tr>
</tbody>
</table>

Template-Based Manipulation
Example

- Operator/algorithm identifies relevant sensor data
- Overlaps template
Template-Based Manipulation Example

- Operator/algorithm identifies relevant sensor data
- Overlaps template
- Selects grasp
Template-Based Manipulation Example

- Operator/algorithm identifies relevant sensor data
- Overlaps template
- Selects grasp
- Performs affordance (see videos)
Template-Based Manipulation
Versatile Manipulation with Unknown Objects
Template-Based Manipulation
Versatile Manipulation with Unknown Objects
Our Contributions (Overview)

Versatile and robust
Perception
• Terrain Modeling

Versatile and robust
(Loco-)Motion
• 3D Footstep Planning in rough terrain

Versatile and robust
Manipulation
• Template-based Manipulation

Efficient
Human-Robot-Interaction
• “Ghost Robot”
• Sliding Autonomy

“Ghost Robot”

- Pre-plan motions with virtual “Ghost Robot”
- Additional capabilities compared to start/goal state visualization in MoveIt! RViz plugin
  - Snap endeffectors to objects
  - Move to stand poses relative to object templates
  - Constrain IK joint limits
  - Send low-bandwidth planning request directly from OCS

https://github.com/team-vigir/vigir_manipulation_planning/tree/master/vigir_ocs_robot_model
Locomotion-Manipulation Pipeline
Locomotion-Manipulation Pipeline

Search for suitable robot pose via inverse reachability query
Generate **footstep plan** to robot pose
Locomotion-Manipulation Pipeline
Example: Hose Task (DRC Trials)
Sliding Autonomy

- Communication constraints
- Limited time
- Complex robot system
- Unstructured environment
- Complex tasks
- Robustness important

Flexible Robot-Operator Collaboration

Motivates high degree of robot autonomy

Motivates high degree of operator support
- **“Flexible Behavior Engine”**
  - Based on SMACH → Hierarchical state machines
  - Adds robot-operator collaboration


https://github.com/team-vigir/flexbe_behavior_engine
FlexBE
Sliding Autonomy

- Behavior runs with explicit Autonomy Level
  - Can be changed any time during execution
- State outcomes define required autonomy
  - High enough → Autonomous execution
  - Too low → Operator confirms or rejects
- Operator can force outcomes any time
FlexBE
Runtime Control
Synergies: Case Studies

- Our software was already applied on following robots:
Synergies: Case Studies

- **Modularity**: Take use of synergies in **Hard- & Software development**

  **Tracker**
  - Full Autonomy
  - Lattice Planning
  - etc…

  **Humanoid**
  - Teleoperation
  - Footstep Planning
  - Balance Control
  - etc…

  **Sliding-Autonomy**
  - SBPL
  - Manipulation
  - Mapping
  - Perception
  - etc…
Synergies
Johnny #5 @RoboCup 2016
Publications

  - Presentation on Thursday 16:30-18:00 (ThPoS.23)

Conclusions

• Humanoid Robots...
  ▪ ...benefit from bipedal locomotion and bimanual manipulation.
  ▪ ...are ideal choice for versatile human tasks in human environments.
  ▪ ...are just robots! Reuse of existing software is highly recommended (e.g. ROS).

• Our contribution:
  ▪ Supervised high-level locomotion and manipulation planning working with constrained communications (bandwidth limitation, delays, packet drops)
  ▪ All presented work is reusable due to modular design
  ▪ Available open source

• Resources:
  ▪ Team ViGIR www.teamvigir.org
  ▪ Team Hector www.teamhector.de
  ▪ Johnny #5 Simulator https://github.com/thor-mang/thor_mang_install