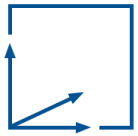
VR Re-Embodiment: Establishing a Control Structure to enable Physic-based Movement of the Human Body in Unity 3D

Konstantin Karas

24.04.2020



Final: Bachelor Informatics: Games Engineering

Supervisor: Sandro Weber

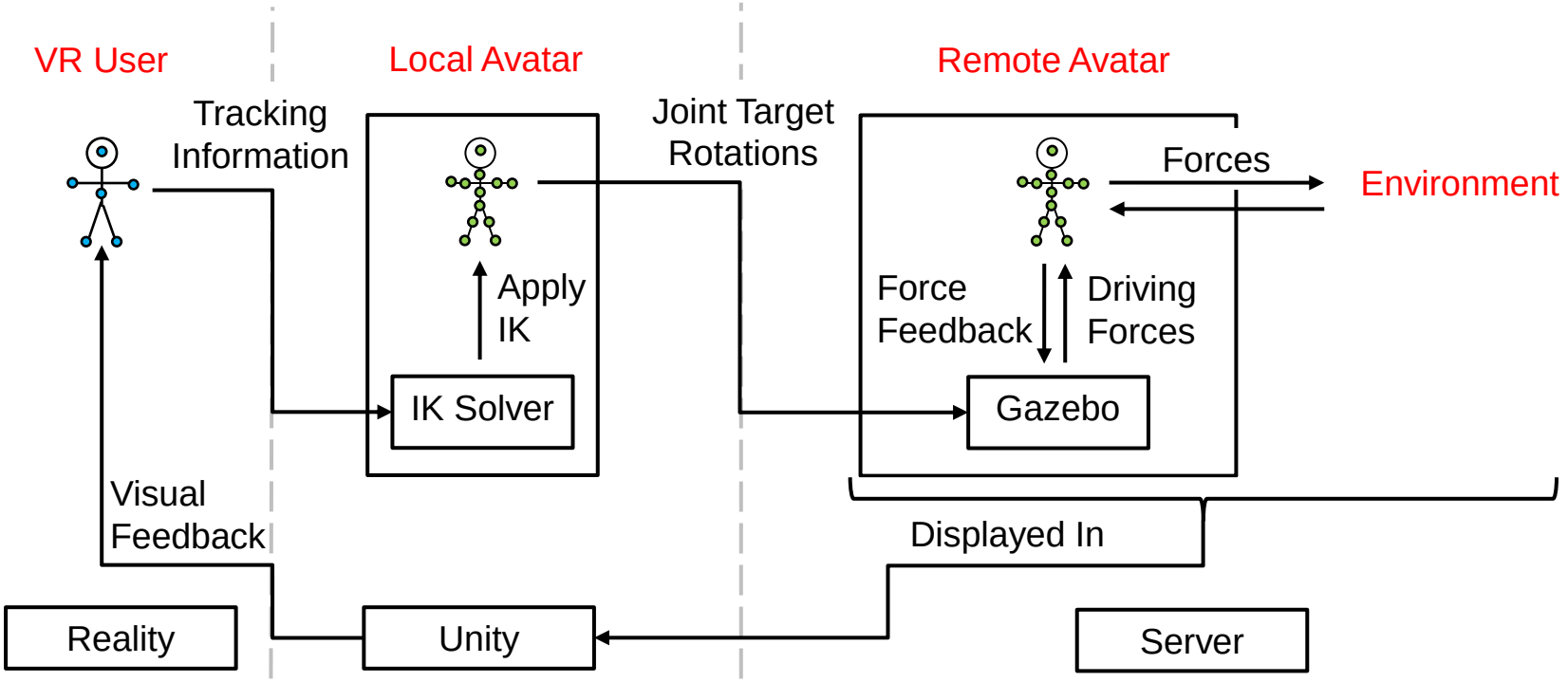
Introduction / Motivation

- **Remote control** of a **robot** through a **human operator**
  - Interaction with environment
  - Virtual training
  - Human embodied in robot

- Sense of Embodiment [1]
  - Sense of Location: “I am at the same place as the robot”
  - **Sense of Agency:** “I am controlling the robot’s movement”
  - Sense of Body-Ownership: “I am the owner of the robot’s body”

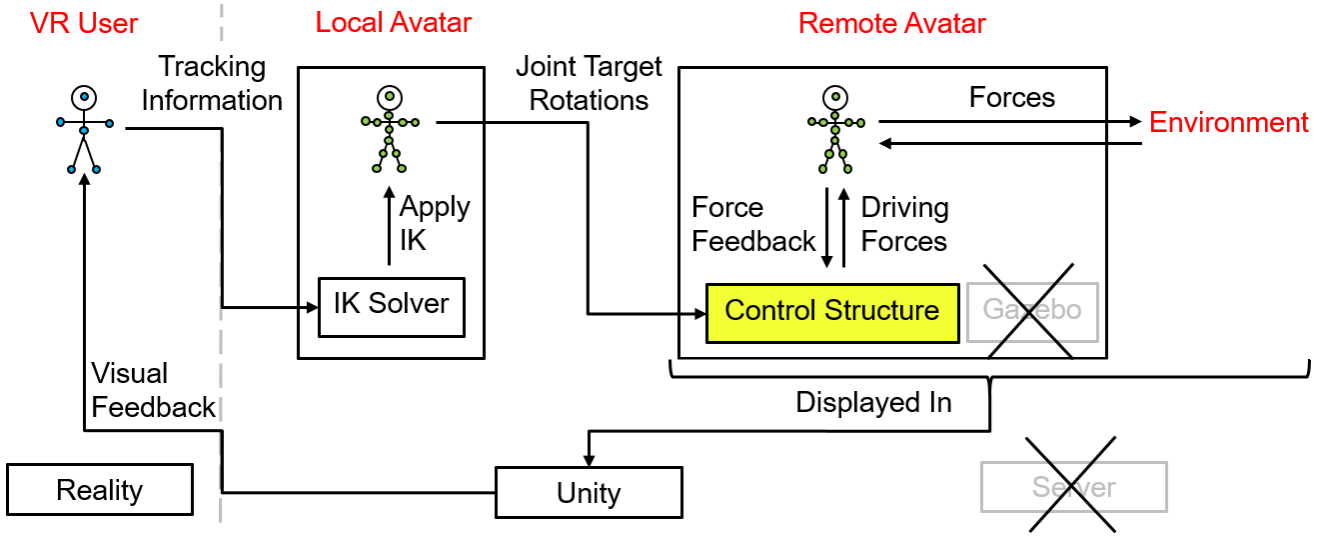
Introduction / Motivation

- Current System [2] (Gazebo [3], HBP Neurorobotics [4])



Problem Description: Issues

- Data has to be sent to the server
  → Information might be outdated
- Solution:



→ **No information sent to server, no latency issues**

Existing Solutions / Related Work

- Many approaches use Inverse Kinematics (IK) and PD control [5, 6]

- Unity Packages:
  - Final IK [7] → blend between animation and IK
  - PuppetMaster [8] → joint + collider setup

*at the time of the thesis

Goals of this Thesis

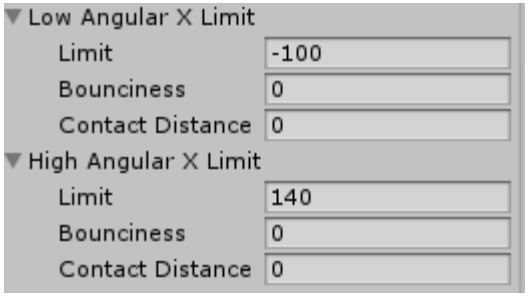
- Providing **physics-based** movement of remote avatar in Unity
- **Independency** from server
- **Fine control** over joints, rigidbodies, colliders
- **Simple setup** and **use**

Proposed Work / Approach

- **How to control the movement of the remote avatar in Unity?**
  - → Rigidbody breaks parenting movement control
  - → Use ConfigurableJoints [9] to connect body parts

Proposed Work / Approach

- **How to control the movement of the remote avatar in Unity?**
  - → Use ConfigurableJoints [9] to connect body parts
    - Angular Limits (values approximated):
      - Low and High X Limit
      - Y Limit
      - Z Limit
    - Target Rotation
    - Angular Drives



Proposed Work / Approach

- **How to control the movement of the remote avatar in Unity?**
  - → Use ConfigurableJoints [9] to connect body parts
    - Angular Limits:
    - Target Rotation
      - Rotation that the joint should turn into and maintain
      - Defined in joint space
      - → Convert **local rotation of local avatar‘s body part** to joint space and assign it
    - Angular Drives

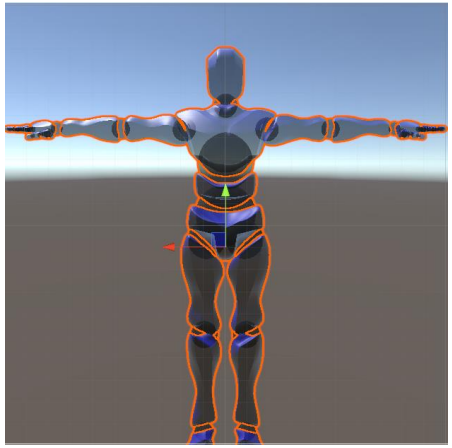
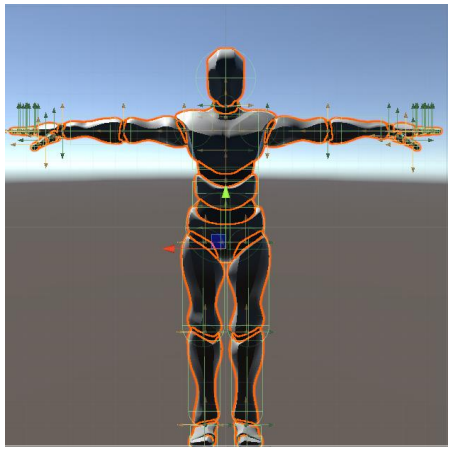
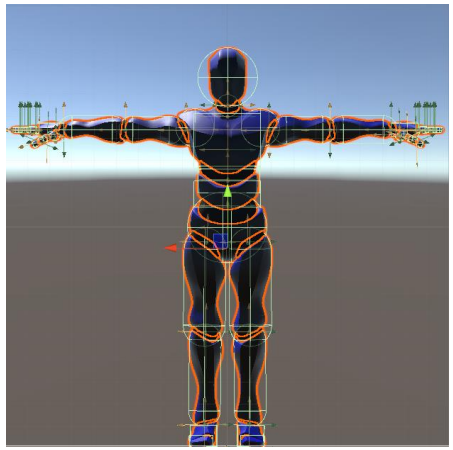
Proposed Work / Approach

- **How to control the movement of the remote avatar in Unity?**
  - → Use ConfigurableJoints [9] to connect body parts
    - Angular Limits:
    - Target Rotation
    - Angular Drives (PD controller):
      - Spring (P value)
      - Damper (D value)
      - Maximum Force

  } for X and YZ axis

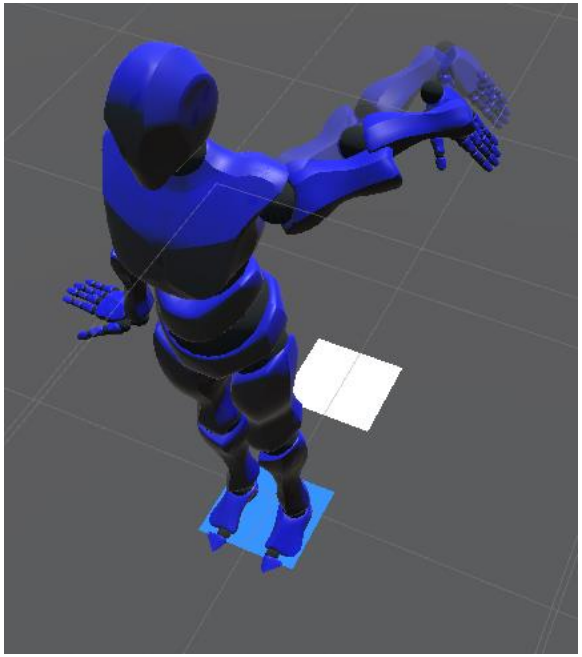
Proposed Work / Approach

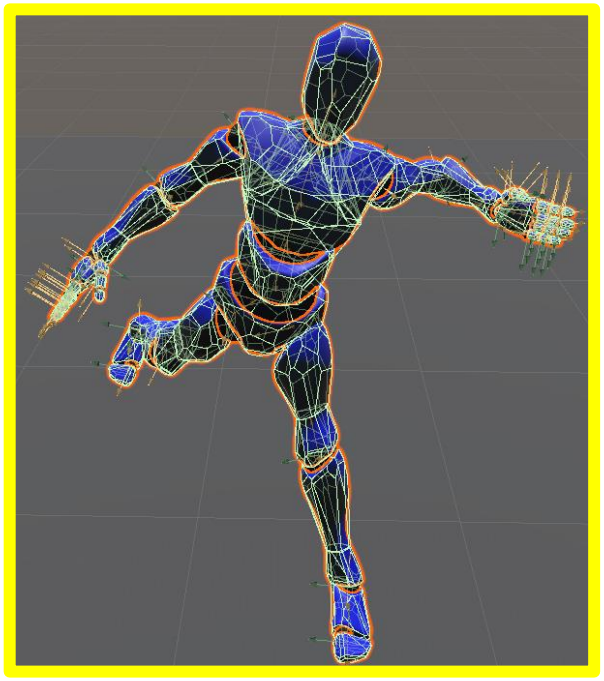
- **How to set up the control structure automatically while still being able to make adjustments to it?**
  - Template avatar for remote avatar configuration
    →Control Structure configured **once** for the template
    →Remote avatar exchangeable
  - Editor window for template configuration

Implementation – Control Structure

- **AvatarManager.cs:** Choose Control Method
  - **Prototype:**
    PD Controllers
  - **ConfigurableJoint Version:**
    Tuned Joints + Colliders



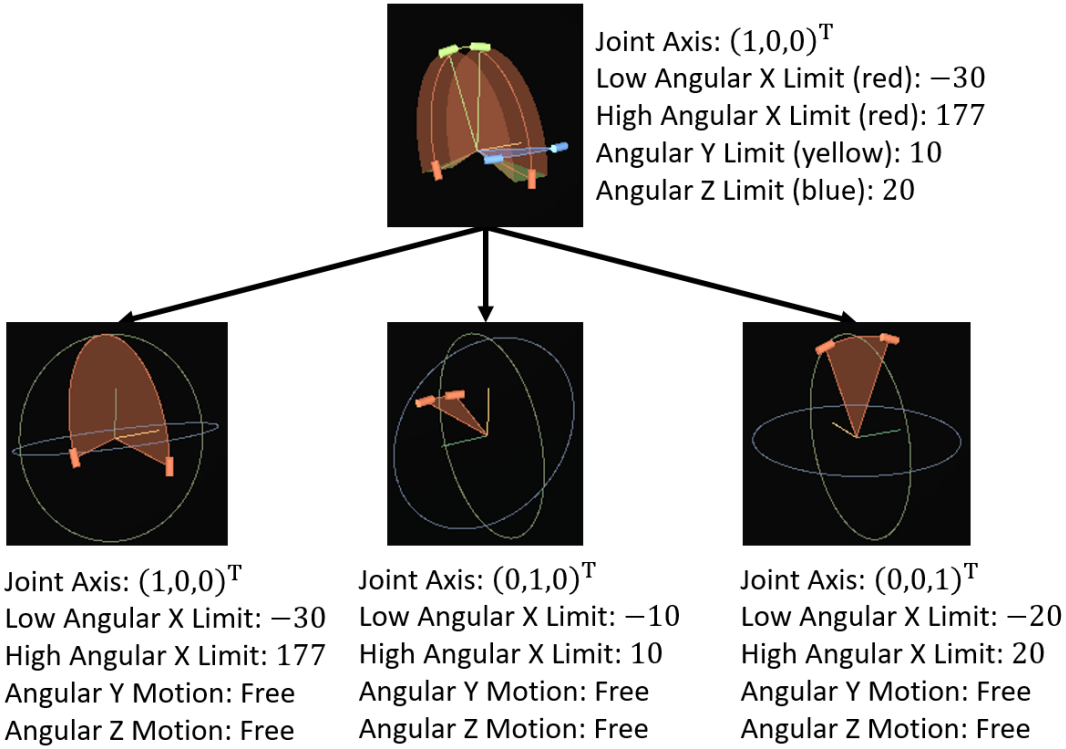


Implementation – Control Structure

- **ConfigJointManager.cs**
  - Configure Control Structure Settings
    - Choose **template** version
      - **Single joint:** 1 ConfigurableJoint per body part
      - **Multiple joint:** 3 ConfigurableJoints per body part (1 for each axis)
    - Choose **Collider** type (simple or mesh)
  - Set **target rotations** of joints in the remote avatar

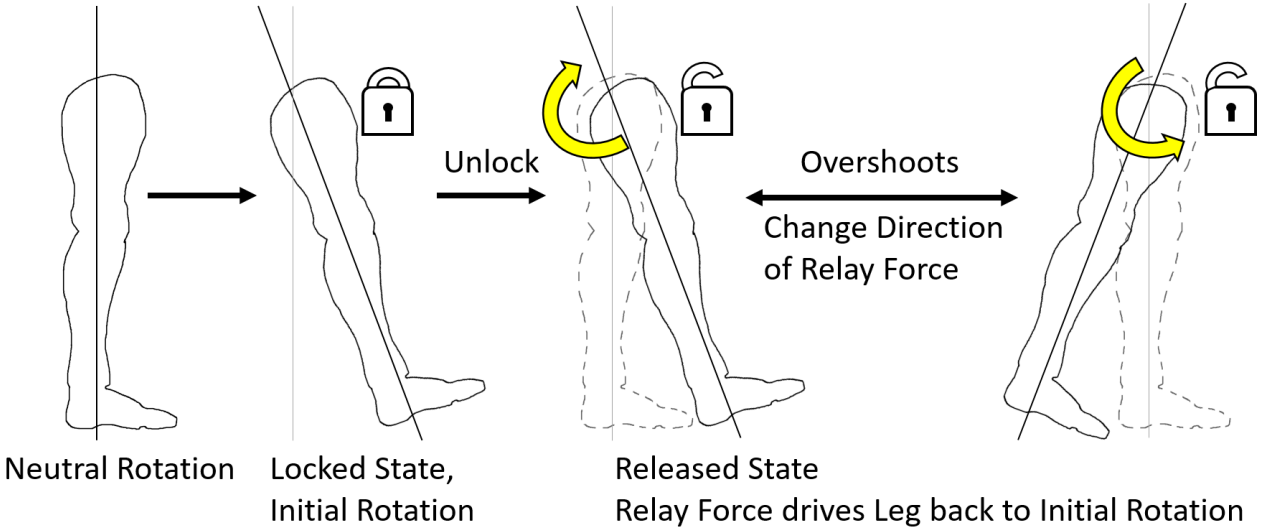
Implementation – Multiple Joint Template

- **JointSetup.cs:** setup and utility
  - **Generate multiple joint template from single joint template:**



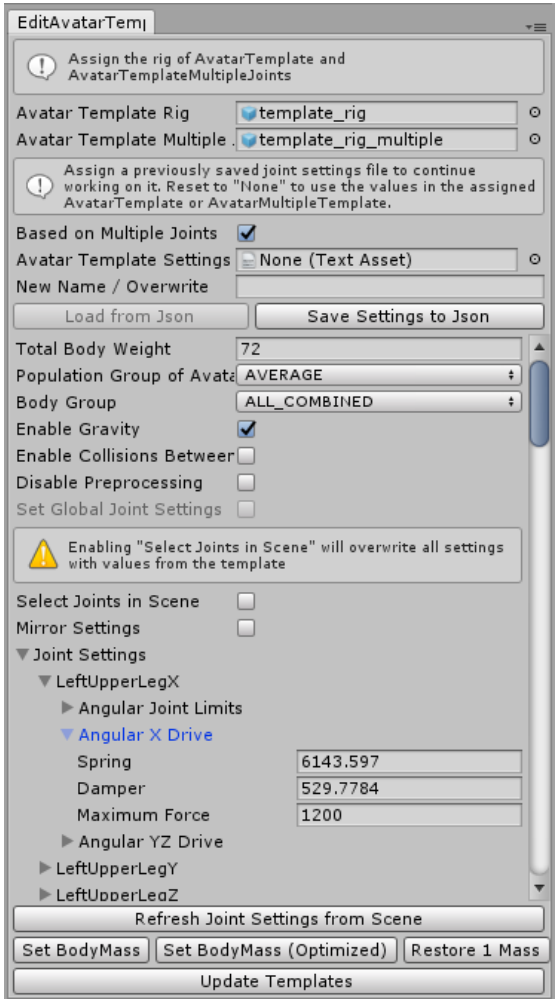
Implementation – Tuning

- Adapt **existing** relay tuning system [10]
- Multiple **mapping functions** added
- Developer can **choose** between either system



Implementation – Editor

- **EditAvatarTemplate.cs:**
  - Choose base template
  - Store changes in **JointSettings.cs**
  - **Filter** through custom groups of **body parts**
  - Set total **body weight**
  - Apply/convert changes to **both** templates
  - **Save & Load** settings



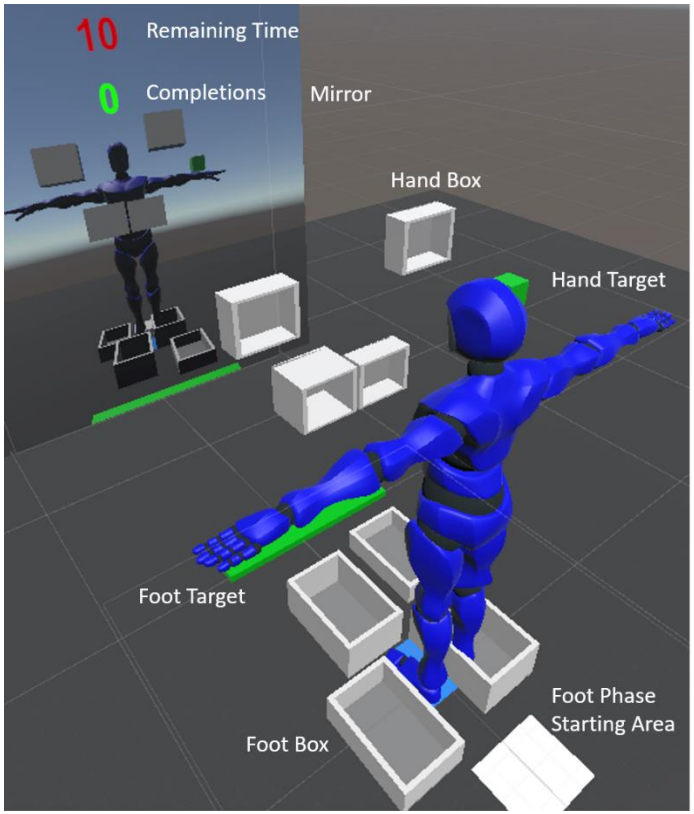
Evaluation

- User Study - Concept:
  - **Artificial latency** (0ms, 125ms, 350ms)
  - Task for hands and feet control → **sense of agency**
  - Evaluation through **standard embodiment questionnaire [11] + time measurements**

	Body Ownership	Agency	Location
Questions	$Q1, Q2, Q3, Q4, Q5$	$Q6, Q7, Q8, Q9$	$Q10, Q11, Q12$
Evaluation	$(Q1 - Q2) - Q3 + (Q4 - Q5)$	$Q6 + Q7 + Q8 - Q9$	$Q10 - Q11 + Q12$
Scale	$-15/15$	$-12/12$	$-9/9$
Sense of Embodiment	$((Ownership/5) * 2 + (Agency/4) * 2 + (Location/3) * 2) / 9$		

Evaluation

- User Study - Concept: Test Environment + Procedure



Evaluation

- Expert Study Feedback
  - Finger control
  - Fingers disconnect
  - Feet control limited (tracker above ankle)

  - Embodiment Questionnaire (0ms latency):

	Body Ownership	Agency	Location
Evaluation	$((-2) - (-3)) - (-3) + (1 - (-3)) = 8$	$3 + 3 + (-2) - (-3) = 7$	$2 - (-3) + (-1) = 4$
Sense of Embodiment	$((8/5) * 2 + (7/4) * 2 + (4/3) * 2) / 9 \approx 1.04$		

→ **Sense of agency** rated as **good** (7), but not perfect (12)

Suggested Future Work

- Conduct user study
- Finer control over fingers desirable
- Joints can disconnect when light/small objects collide with heavy/immovable objects

Conclusion

- Fully functional **control structure**
- **Tuning** through tuning system
- Custom **editor** for joints
- **Save & Load** tuning and joints‘ settings
- Complete **user study** concept + test environment

List of References

1. K. Kilteni, R. Groten and M. Slater. ‘The Sense of Embodiment in Virtual Reality’. In: Presence: Teleoperators and Virtual Environments 21.4 (2012), pp. 373–387. issn: 1054-7460. doi: 10.1162/PRES_a_00124
2. S.Weber and G. Klinker. ‘VR Re-Embodiment in the Neurorobotics Platform’. In: Mensch und Computer 2019 - Workshopband (Hamburg, Germany). Bonn: Gesellschaft für Informatik e.V., 2019. doi: 10.18420/muc2019-ws-585
3. OSRF. Gazebo. url: http://gazebosim.org/ (visited on 23/04/2020).
4. HBP. Neurorobotics Platform. url: https://neurorobotics.net/ (visited on 23/04/2020).
5. M. Oshita and A. Makinouchi. ‘A Dynamic Motion Control Technique for Human–like Articulated Figures’. In: Computer Graphics Forum 20.3 (2001), pp. 192–203. issn: 1467-8659. doi: 10.1111/1467-8659.00512. url: https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-8659.00512.

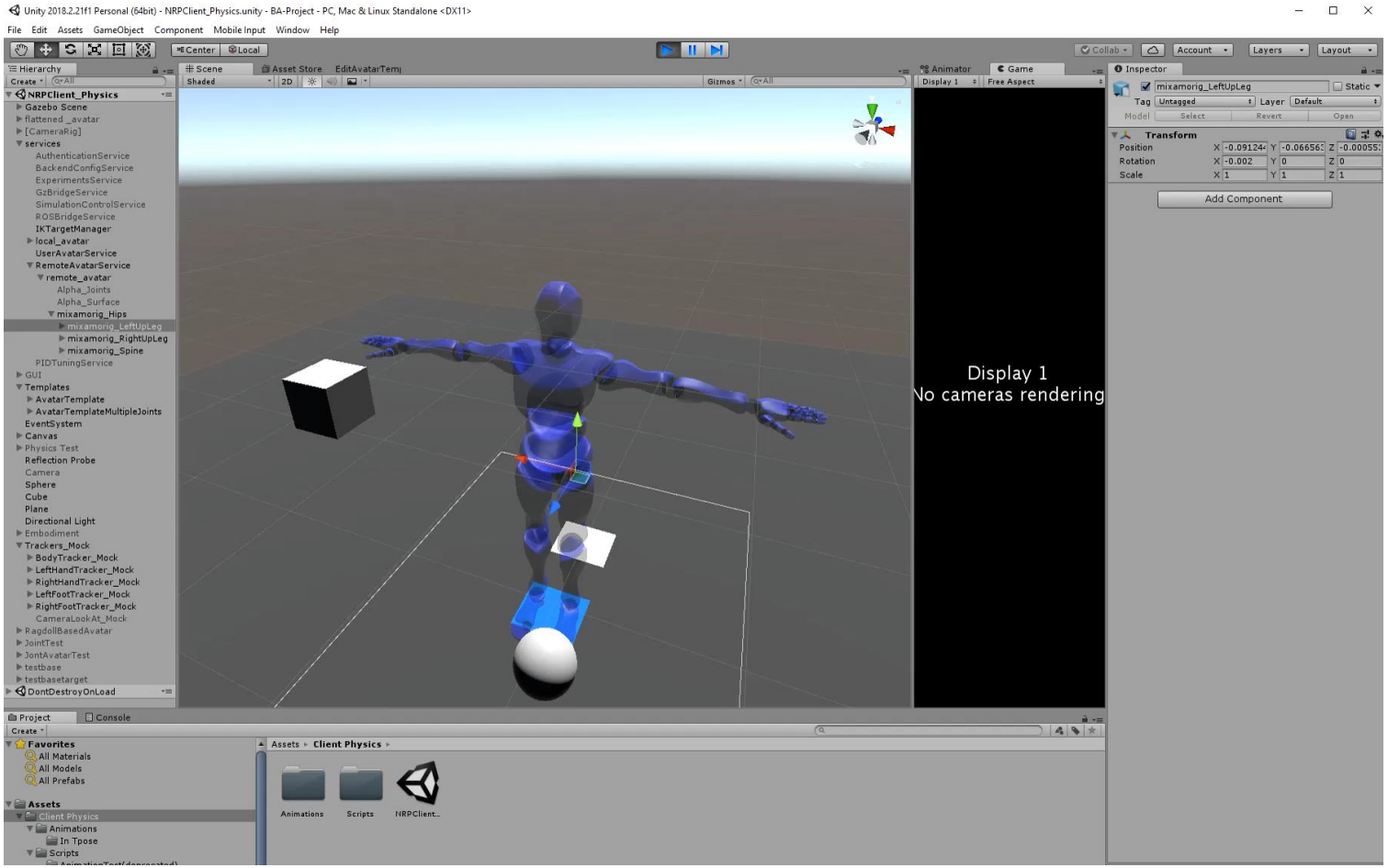
List of References

6. V. B. Zordan and J. K. Hodgins. ‘Motion Capture-Driven Simulations That Hit and React’. In: Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (San Antonio, Texas, USA). SCA ’02. New York, NY, USA: Association for Computing Machinery, 2002, pp. 89–96. isbn: 1581135734. doi: 10.1145/545261.545276. url: https://doi.org/10.1145/545261.545276.
7. RootMotion. Final IK | Animation | Unity Asset Store. 2019. url: https://assetstore.unity.com/packages/tools/animation/final-ik-14290 (visited on 23/04/2020).
8. RootMotion. PuppetMaster | Physics | Unity Asset Store. 2019. url: https://assetstore.unity.com/packages/tools/physics/puppetmaster-48977 (visited on 23/04/2020).

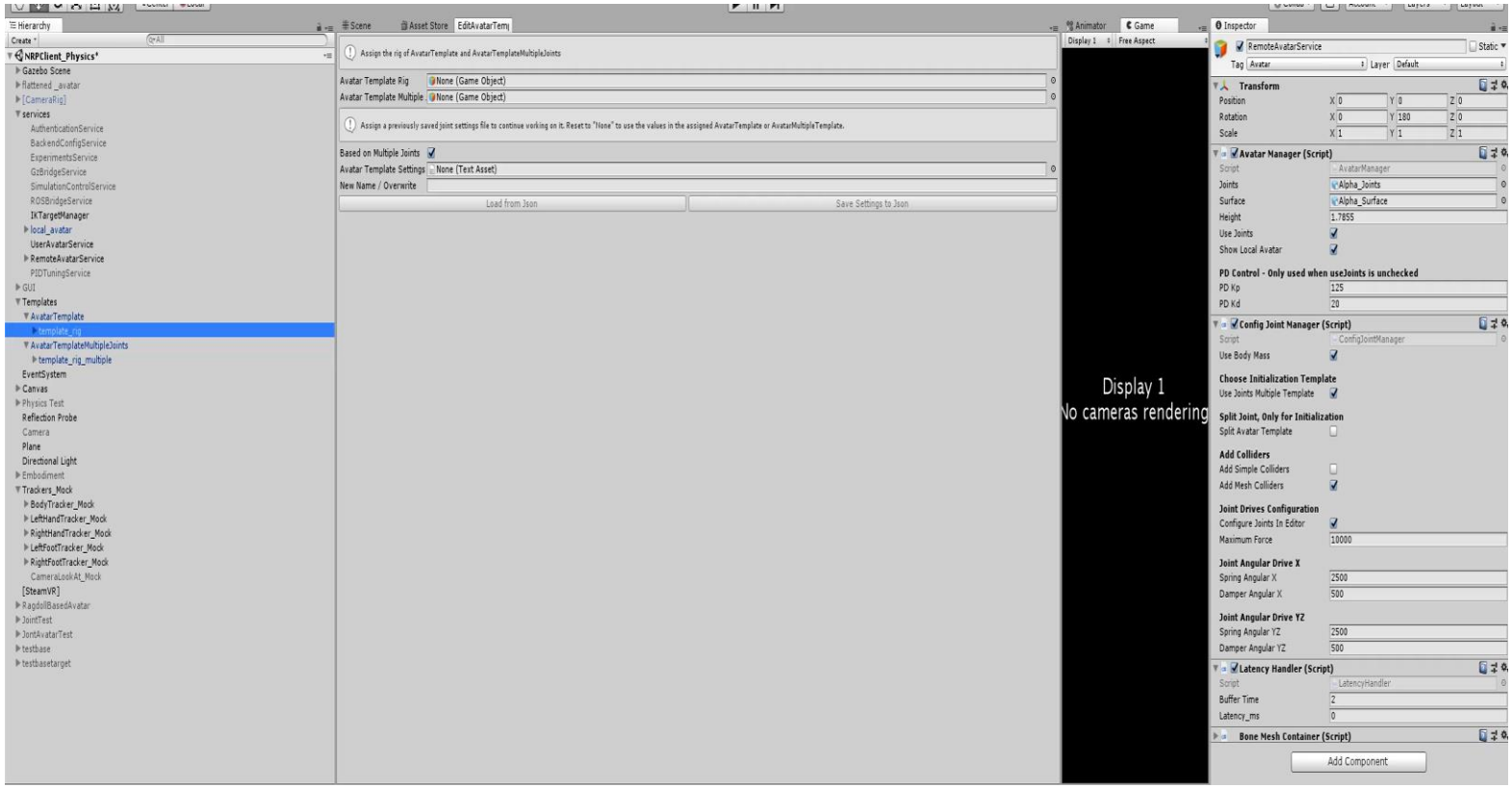
List of References

9. NVIDIA. Joints — NVIDIA PhysX SDK 3.4.0 Documentation. 2017. url: https://docs.nvidia.com/gameworks/content/gameworkslibrary/physx/guide/Manual/Joints.html#d6-joint (visited on 23/04/2020).
10. M. Webel. PID-Tuning Framework for Remotely Operated Humanoid Robots. 2019. url: https://wiki.tum.de/display/infar/MA%3A+PID-Tuning+Framework+for+Remotely+Operated+Humanoid+Robots (visited on 23/04/2020).
11. M. Gonzalez-Franco and T. C. Peck. 'Avatar Embodiment. Towards a Standardized Questionnaire'. In: Frontiers in Robotics and AI 5 (2018), p. 74. issn: 2296-9144. doi: 10.3389/frobt.2018.00074. url: https://www.frontiersin.org/articles/10.3389/frobt.2018.00074/pdf.

Movement Demo

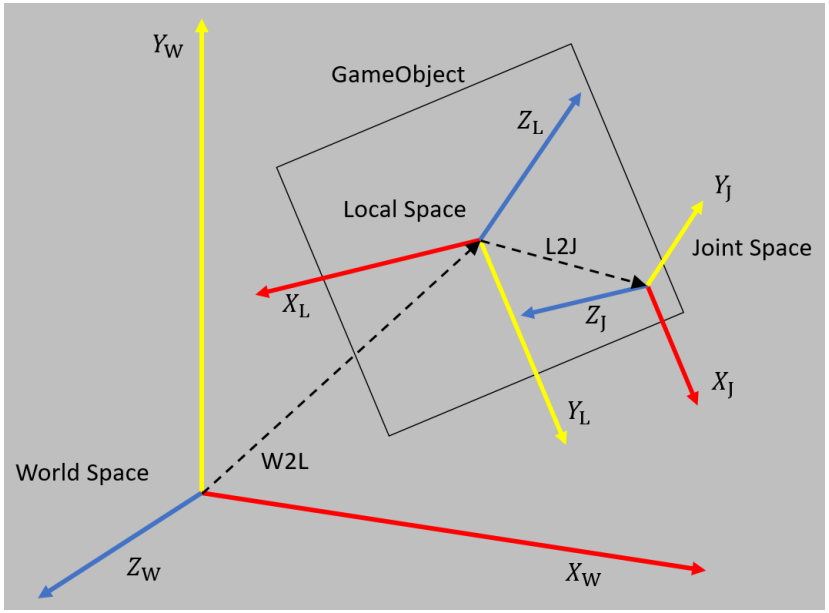


Editor Demo


Hierarchy
Scene
Asset Store
Animator
Game
Inspector
NRPClient_Physics*
Gazebo Scene
services
Templates
AvatarTemplate
AvatarTemplateMultipleJoints
EventSystem
Canvas
Physics Test
Reflection Probe
Camera
Plane
Directional Light
Trackers_Mock
Assign the rig of AvatarTemplate and AvatarTemplateMultipleJoints
Avatar Template Rig
Avatar Template Multiple
None (Game Object)
Assign a previously saved joint settings file to continue working on it. Reset to "None" to use the values in the assigned AvatarTemplate or AvatarMultipleTemplate.
Based on Multiple Joints
Avatar Template Settings
None (Text Asset)
New Name / Overwrite
Load from Json
Save Settings to Json
Display 1
No cameras rendering
RemoteAvatarService
Transform
Position
Rotation
Scale
Avatar Manager (Script)
Joints
Alpha_Joints
Surface
Alpha_Surface
Height
1.7855
Use Joints
Show Local Avatar
PD Control - Only used when useJoints is unchecked
PD Kp
125
PD Kd
20
Config Joint Manager (Script)
Use Body Mass
Choose Initialization Template
Use Joints Multiple Template
Split Joint, Only for Initialization
Split Avatar Template
Add Colliders
Add Simple Colliders
Add Mesh Colliders
Joint Drives Configuration
Configure Joints In Editor
Maximum Force
10000
Joint Angular Drive X
Spring Angular X
2500
Damper Angular X
500
Joint Angular Drive YZ
Spring Angular YZ
2500
Damper Angular YZ
500
Latency Handler (Script)
Buffer Time
2
Latency_ms
0
Bone Mesh Container (Script)
Add Component

Proposed Work / Approach

- **How to control the movement of the remote avatar in Unity?**
  - → Use ConfigurableJoints [8] to connect body parts
    - Joint Space
      - Primary axis (X_J)
      - Secondary axis (Y_J)
    - DoFs
    - Angular Limits
    - Target Rotation
    - Angular Drives



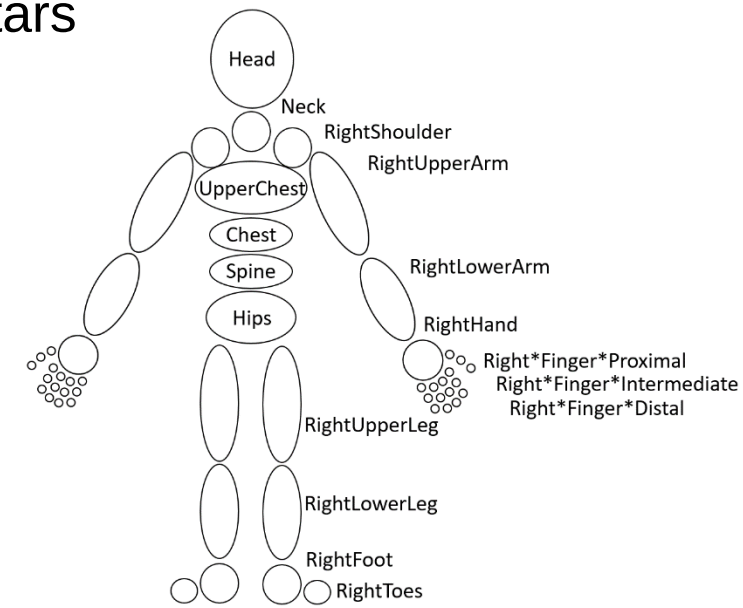
Proposed Work / Approach

- **How to control the movement of the remote avatar in Unity?**
  - → Use ConfigurableJoints [8] to connect body parts
    - Joint Space
    - DoFs: Angular X/Y/Z
      - Free: Unrestricted
      - Limited: Restricted by limits
      - Locked: Rotation impossible
    - Angular Limits
    - Target Rotation
    - Angular Drives

Proposed Work / Approach

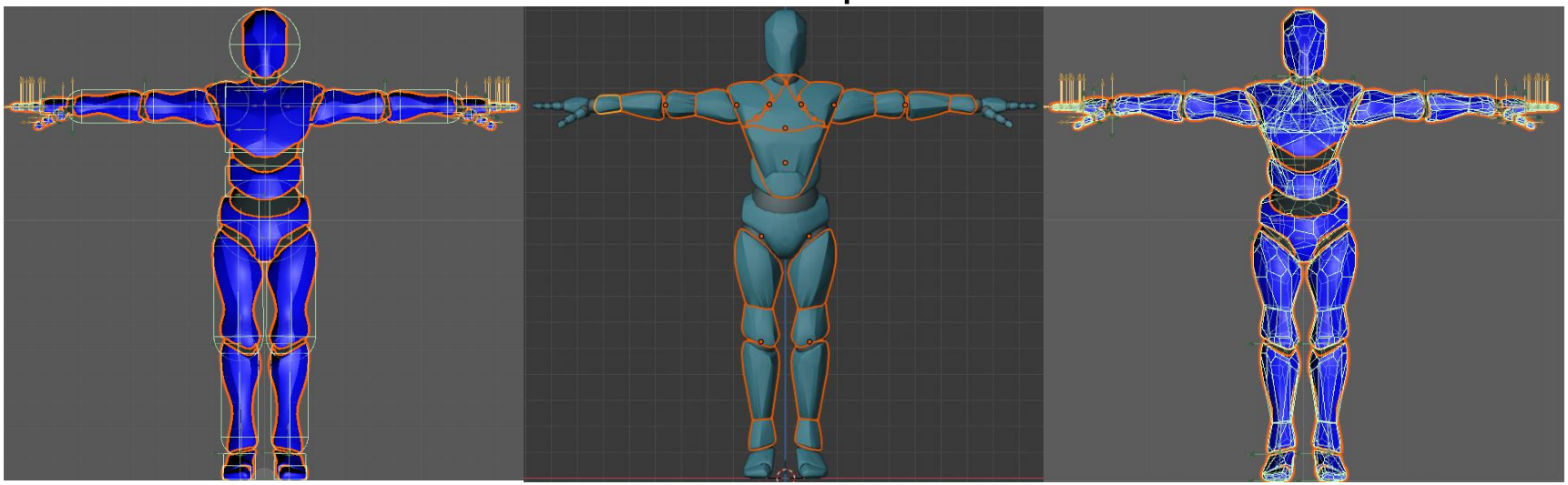
- **How to set up the control structure automatically while still being able to make adjustments to it?**
  → HumanBodyBones:
    - Internal value for key body parts of the human body
    - Use to navigate through avatars

  → Objects‘ names not important



Implementation – Collider Types

- **JointSetup.cs:** setup and utility
  - Copy **simple Colliders** from template or assign **MeshColliders** from storage:
    - Not both active at once
    - Can be **toggled** at runtime
    - Meshes **custom** made for specific model



Fingers Disconnect

- Physics engine has problems dealing with
  - Large weight differences
  - Small volume of colliders