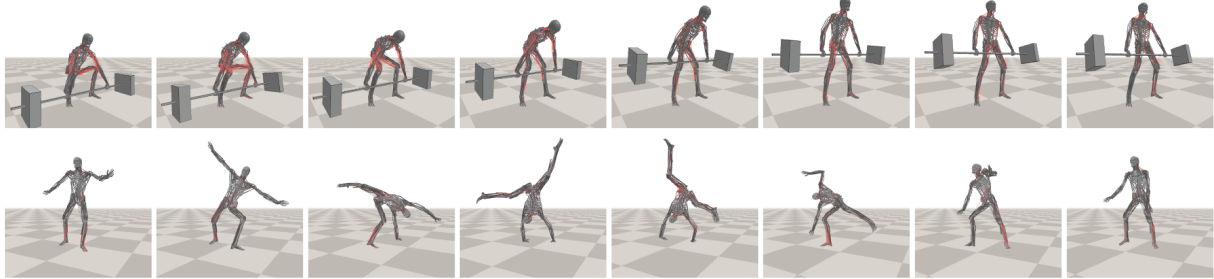

MASS(Muscle-Actuated Skeletal System)



Abstract

This code implements a basic simulation and control for full-body **Musculoskeletal** system. Skeletal movements are driven by the actuation of the muscles, coordinated by activation levels. Interfacing with python and pytorch, it is available to use Deep Reinforcement Learning(DRL) algorithm such as Proximal Policy Optimization(PPO).

Publications

Seunghwan Lee, Kyoungmin Lee, Moonseok Park, and Jehee Lee Scalable Muscle-actuated Human Simulation and Control, ACM Transactions on Graphics (SIGGRAPH 2019), Volume 37, Article 73.

Project Page : <http://mrl.snu.ac.kr/research/ProjectScalable/Page.htm>

Youtube : <https://youtu.be/a3jfyJ9JVeM>

Paper : <http://mrl.snu.ac.kr/research/ProjectScalable/Paper.pdf>

How to install

Install TinyXML, Eigen, OpenGL, assimp, Python3, etc...

```
1 sudo apt-get install libtinyxml-dev libeigen3-dev libxi-dev libxmu-dev
   freeglut3-dev libassimp-dev libpython3-dev python3-tk python3-numpy
   virtualenv ipython3 cmake-curses-gui libbullet-dev pybind11-dev
```

Install DART 6.8

Please refer to <http://dartsim.github.io/> (Install version 6.8)

Manual from DART(http://dartsim.github.io/install_dart_on_ubuntu.html)

Install PIP things

You should first activate virtualenv.

```
1 virtualenv /path/to/venv --python=python3
2 source /path/to/venv/bin/activate
```

- pytorch(<https://pytorch.org/>)
- numpy, matplotlib

```
1 pip3 install numpy matplotlib ipython
```

How to compile and run

Resource

Our system require a reference motion to imitate. We provide sample references such as walking, running, and etc...

To learn and simulate, we should provide such a meta data. We provide default meta data in /data/metadata.txt. We parse the text and set the environment. Please note that the learning settings and the test settings should be equal.(metadata.txt should not be changed.)

Compile and Run

```
1 mkdir build
2 cd build
3 cmake ..
4 make -j8
```

- Run Training

```
1 cd python
2 source /path/to/virtualenv/
3 python3 main.py -d ../data/metadata.txt
```

All the training networks are saved in /nn folder.

- Run UI

```
1 source /path/to/virtualenv/
2 ./render/render ../data/metadata.txt
```

-
- Run Trained data

```
1 source /path/to/virtualenv/  
2 ./render/render ../data/metadata.txt ../nn/xxx.pt ../nn/xxx_muscle.pt
```

If you are simulating with the torque-actuated model,

```
1 source /path/to/virtualenv/  
2 ./render/render ../data/metadata.txt ../nn/xxx.pt
```

Model Creation & Retargeting (This module is ongoing project.)

This requires Maya and MotionBuilder.

There is a sample model in data/maya folder that I generally use. Currently if you are trying to edit the model, you have to make your own export maya-python code and xml writer so that the simulation code correctly read the musculoskeletal structure. There is also a rig model that is useful to retarget a new motion.