Data Driven Computer Animation

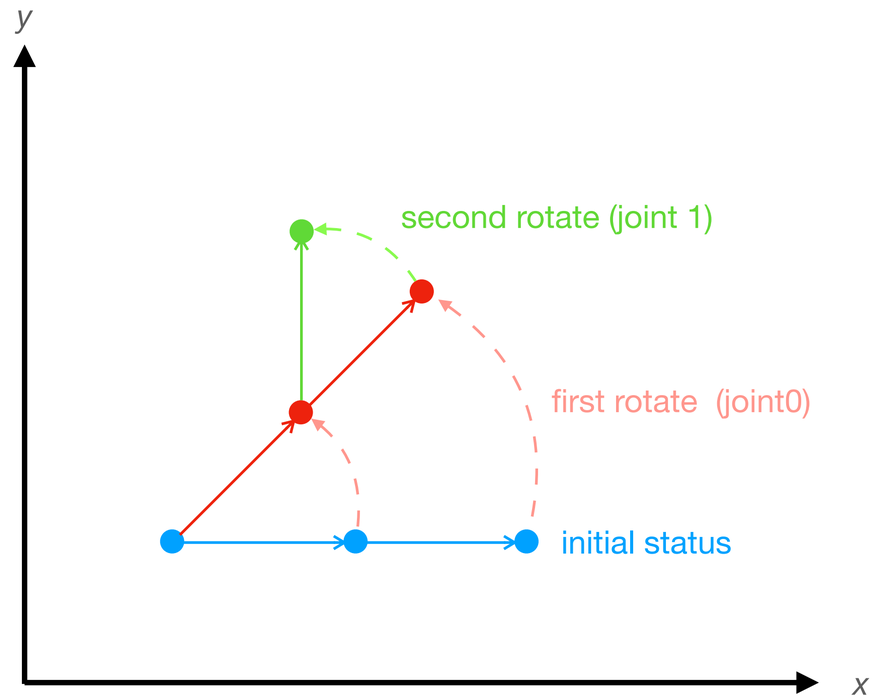
HKU COMP 3360

Tutorial 3 - Motion Processing and ML Setting Up

Prof. Taku Komura
TA: Shi Mingyi (myshi@cs.hku.hk)

SECTION 2A, 2021

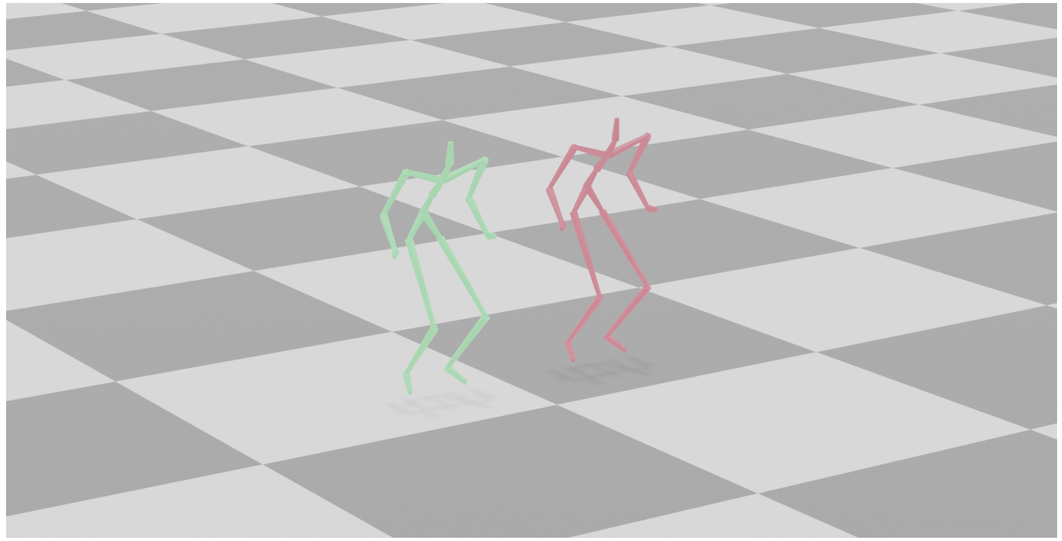
Review on Assignment 1



Rotations in a joint chain

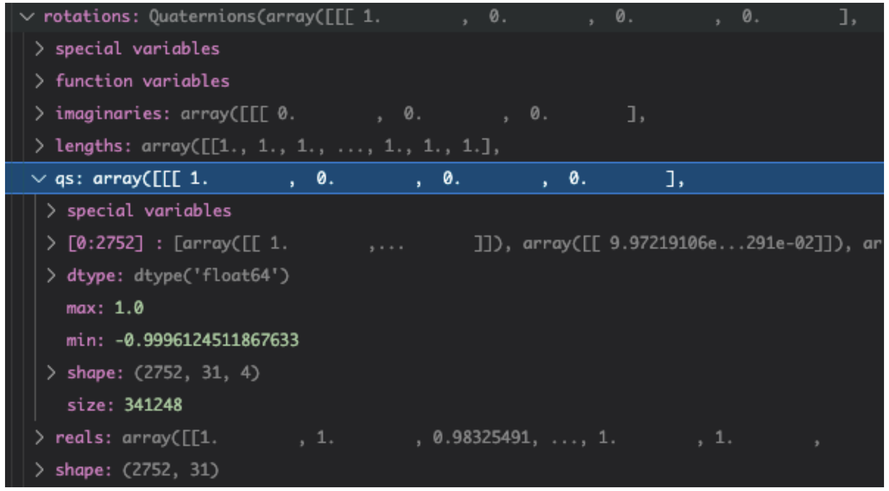
Tutorial 3 - Agenda

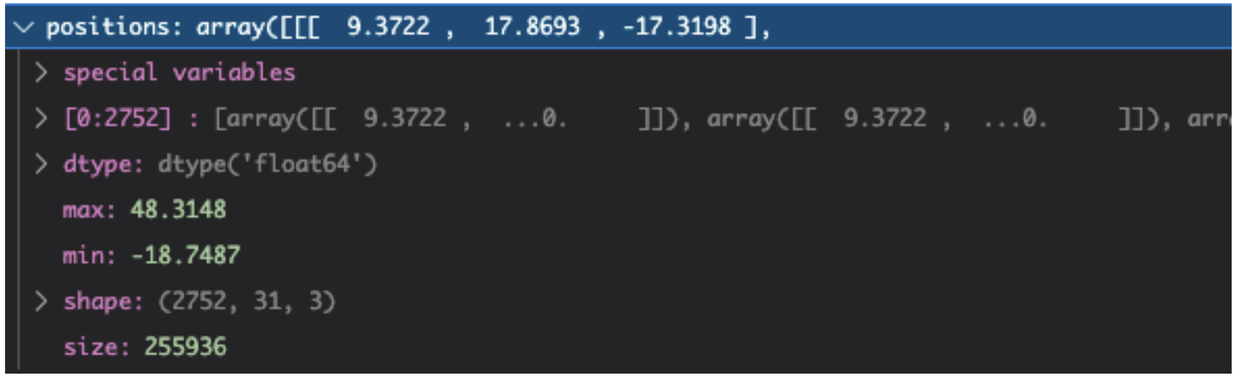
- Motion Processing, Programming Guidance
- Motion Interpolation/concatenation
- Deep Learning Framework Basic(PyTorch)

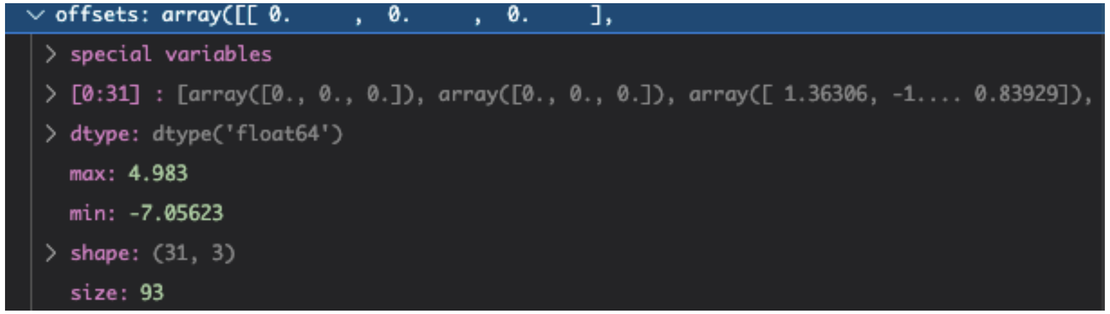


(Under visual studio code programming environment)

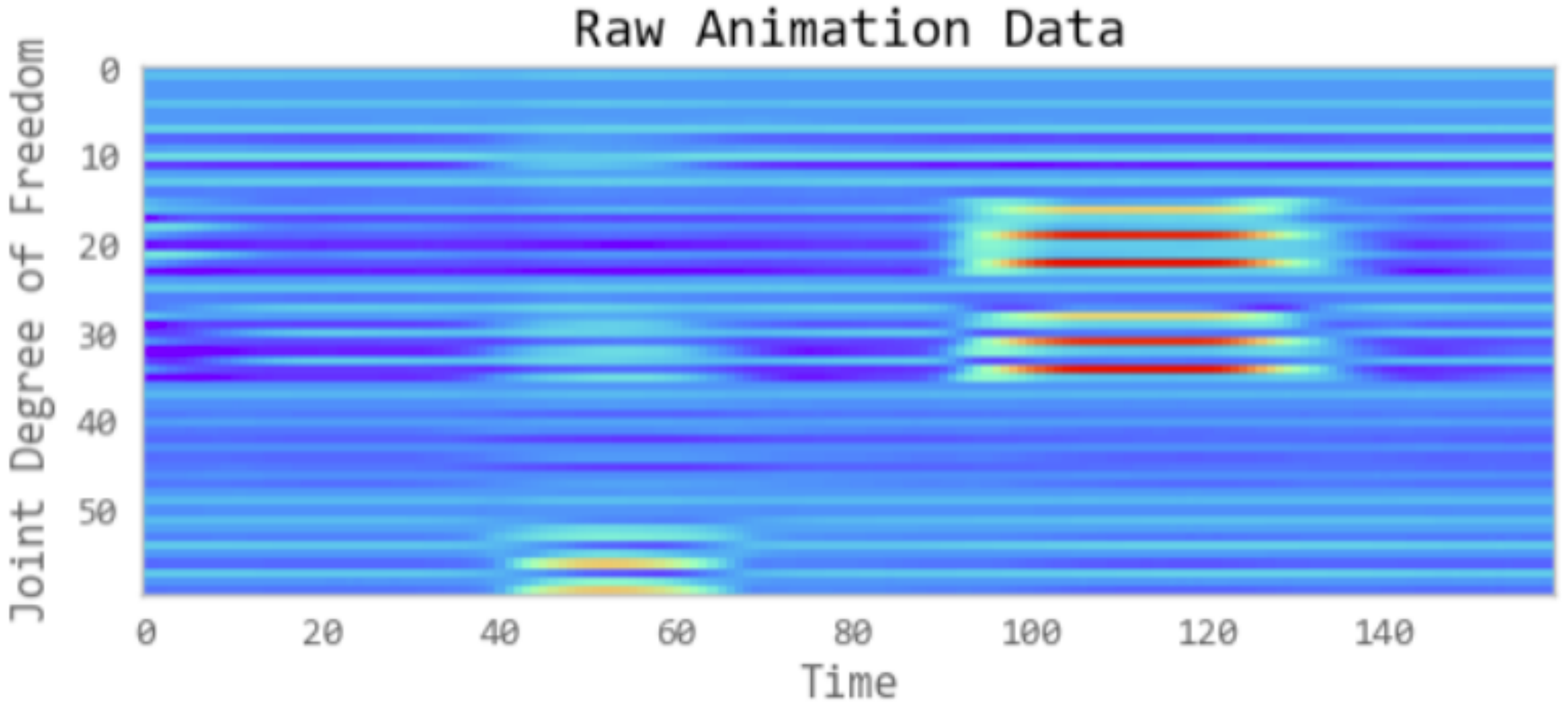
Motion Data Observation



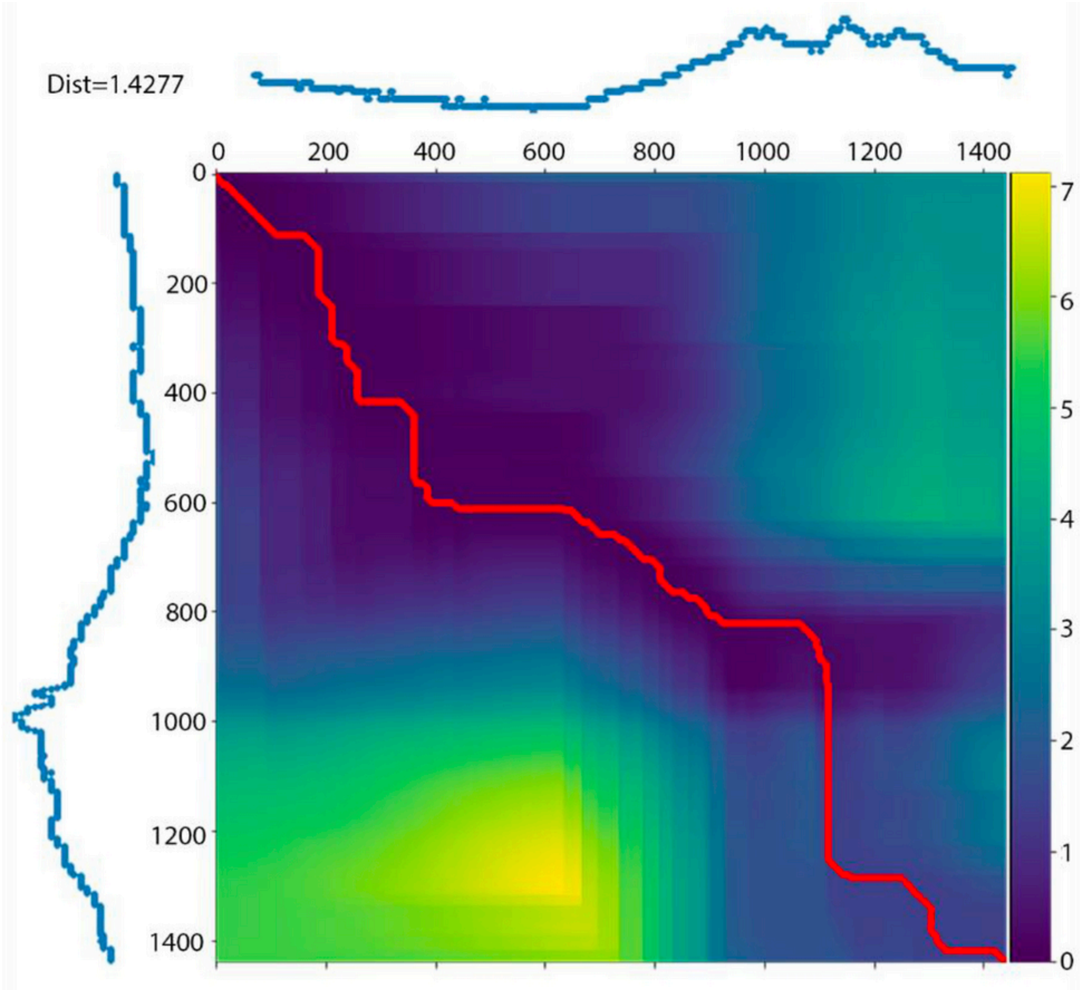




Visualization: Red/Blue Map



Visualization: Dynamic Time Warping



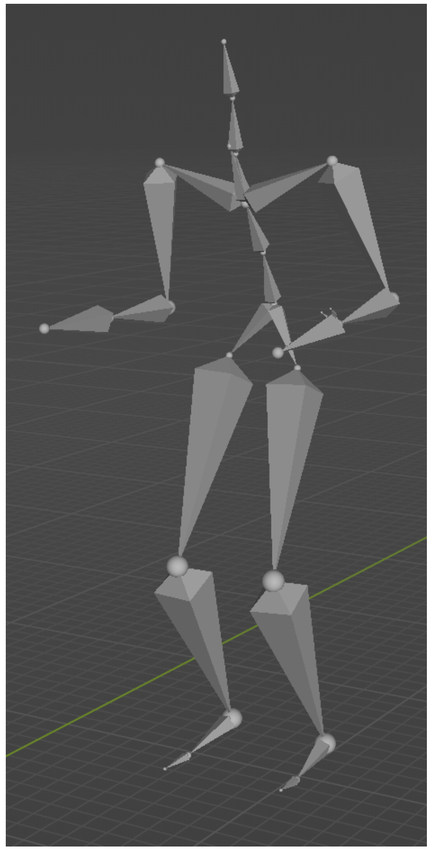
More Operations for Motion Data

There is a motion data with 2000 frames

- Downsampling frames -> change fps
  - *motion = motion[np.arange(0, 2000, 2)]*
- Replay motion
  - motion = np.concatenate(motion, motion[::-1])
- Concat different motions
  - motion = np.concatenate(motion1, motion2)

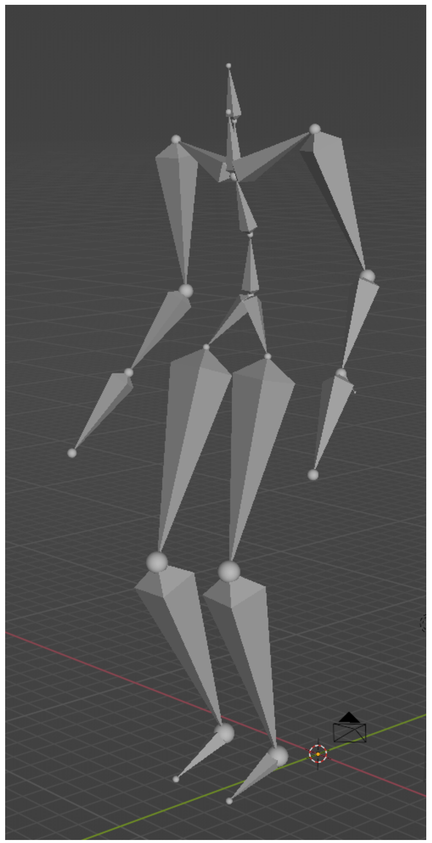
Oops

The last frame of motion1 is different with the first frame or motion2



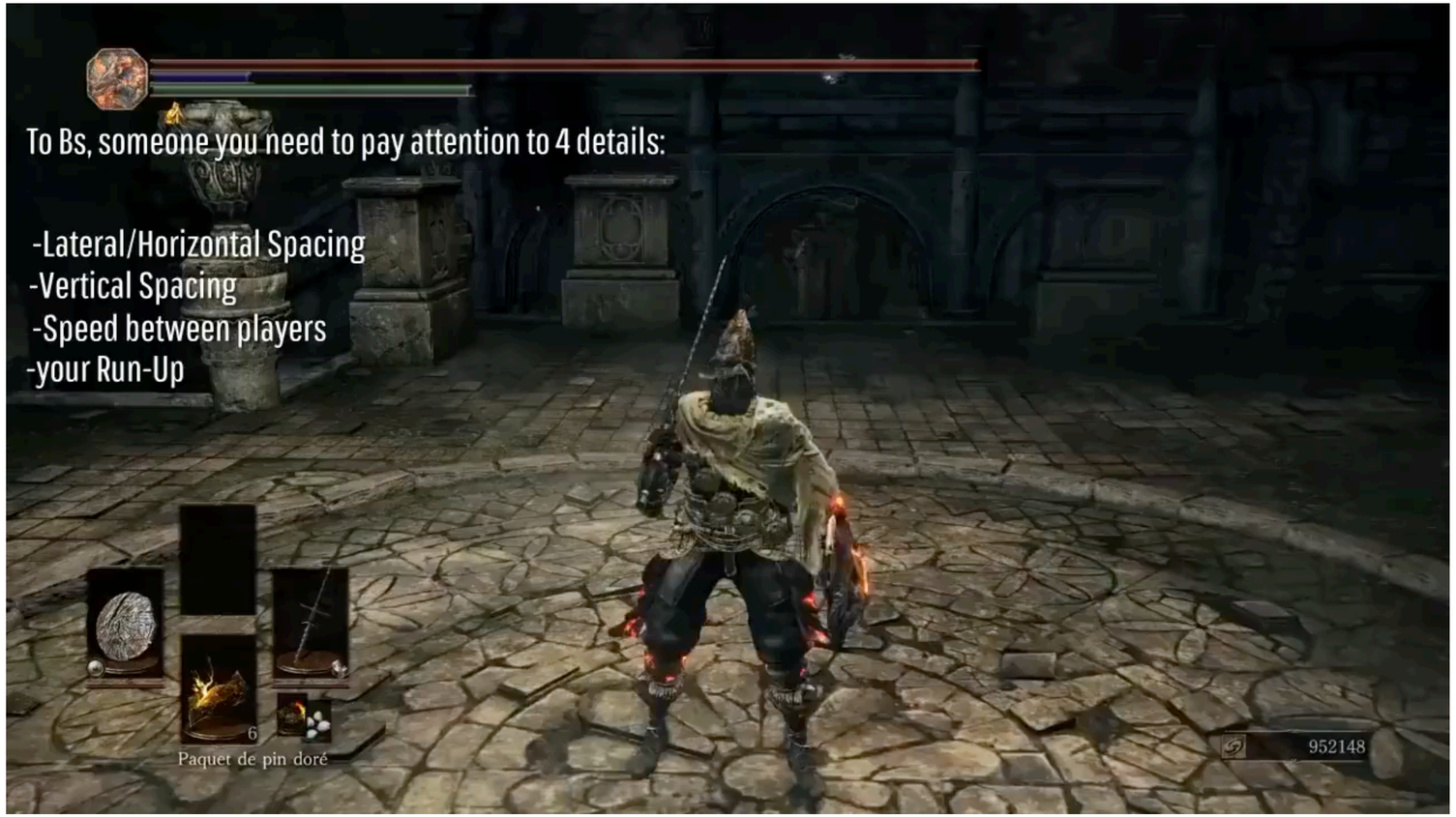
last frame of motion1

?

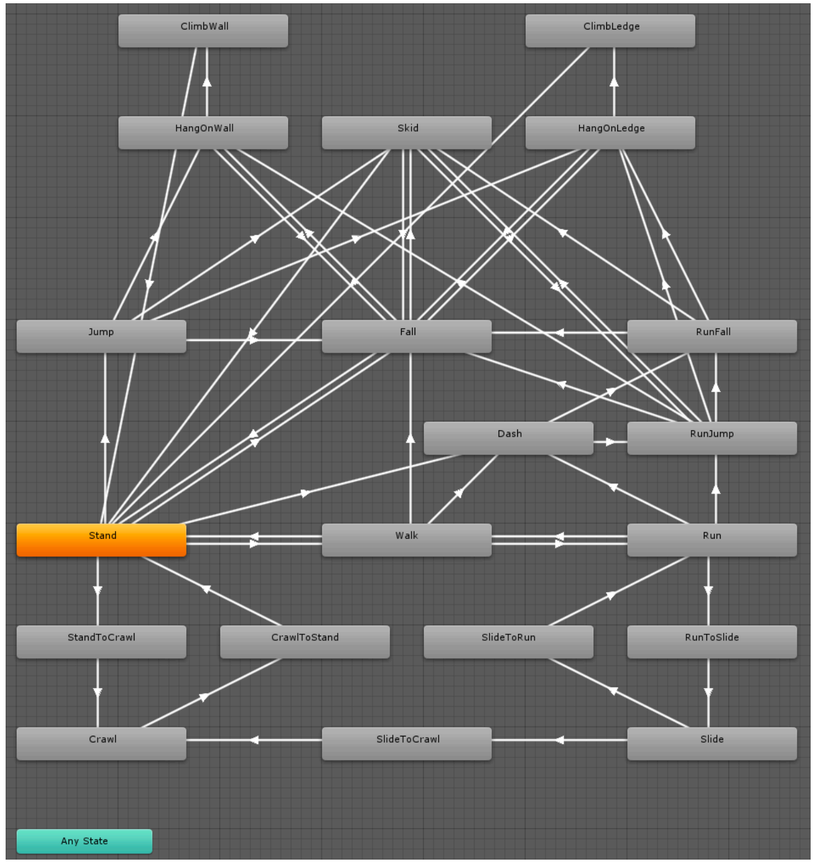


first frame of motion2

Common problem



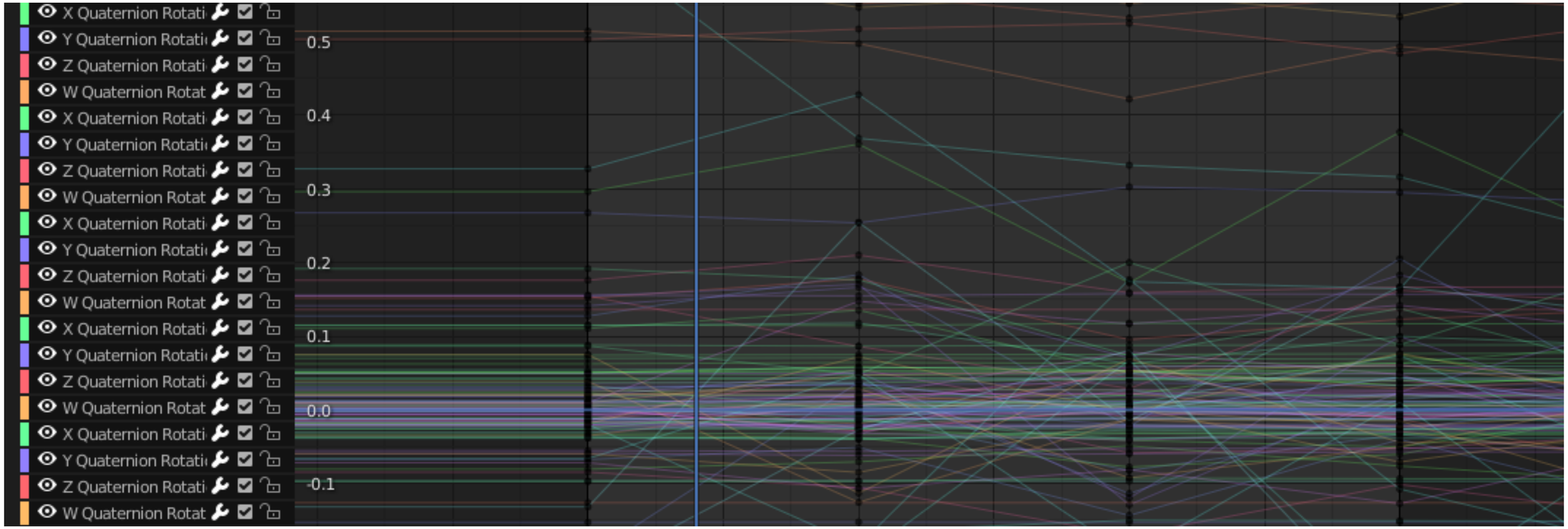
Controlling system is a combination for lots of motion clips



Animation State Machine

Motion Interpolation

You tried that in the assignment1.quiz1

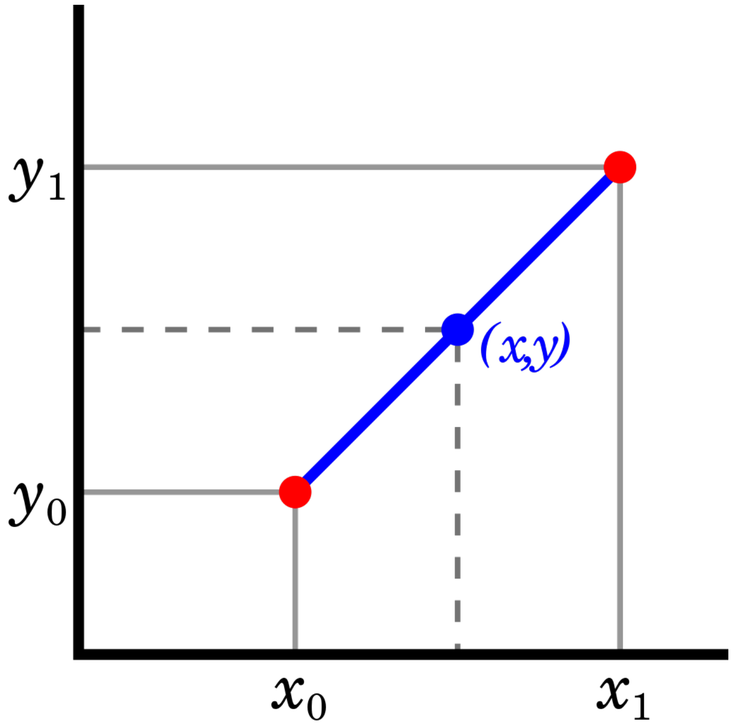


The interpolation with Quaternion

Linear Interpolation

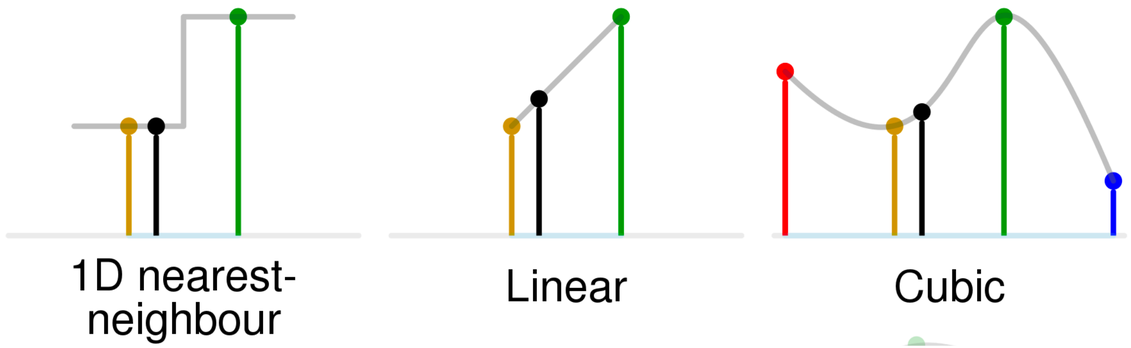
- 1, 2, 3, 4, 5, 6, ?, ?, ?, 10, 11 can you fill it?

- 1, ?, ?, ?, ?, ?, ? ?, ?, ?, 11 can you fill it?



Linear interpolation, refer to wiki

Advanced Interpolation



Task1: Try to use 3 interpolations methods and report its performance

Task 1: Motion Interpolation

- motion_interpolation.py (20%)
    - Implement 2 interpolation interface (any 3rd library is allowed)
    - Report the performance when OFFSET is 24, 48, 72, 96, 124 in a table
        - Visual performance
        - Errors between your interpolation and ground truth
- motion_concatenation.py (30%)
    - Extend you interpolation function into blending task, try your best to shift the difference between last_motion1 and first_motion2
    - Change the last frame index for motion1, test your function and solve the new coming problems

Deep Learning Framework Basic

- Beginner friendly version

- What is deep learning? (officially)

Deep learning

From Wikipedia, the free encyclopedia

For deep versus shallow learning in educational psychology, see Student approaches to learning. For more information, see Artificial neural network.

Deep learning (also known as **deep structured learning**) is part of a broader family of machine learning methods based on artificial neural networks with representation learning. Learning can be supervised, semi-supervised or unsupervised.[2][3][4]

Deep-learning architectures such as deep neural networks, deep belief networks, deep reinforcement learning, recurrent neural networks and convolutional neural networks have been applied to fields including computer vision, speech recognition, natural language processing, machine translation, bioinformatics, drug design, medical image analysis, climate science, material inspection and board game programs, where they have produced results comparable to and in some cases surpassing human expert performance.[5][6][7][8]

Artificial neural networks (ANNs) were inspired by information processing and distributed communication nodes in biological systems. ANNs have various differences from biological brains. Specifically, artificial neural networks tend to be static and symbolic, while the biological brain of most living organisms is dynamic (plastic) and analogue.[9][10][11]

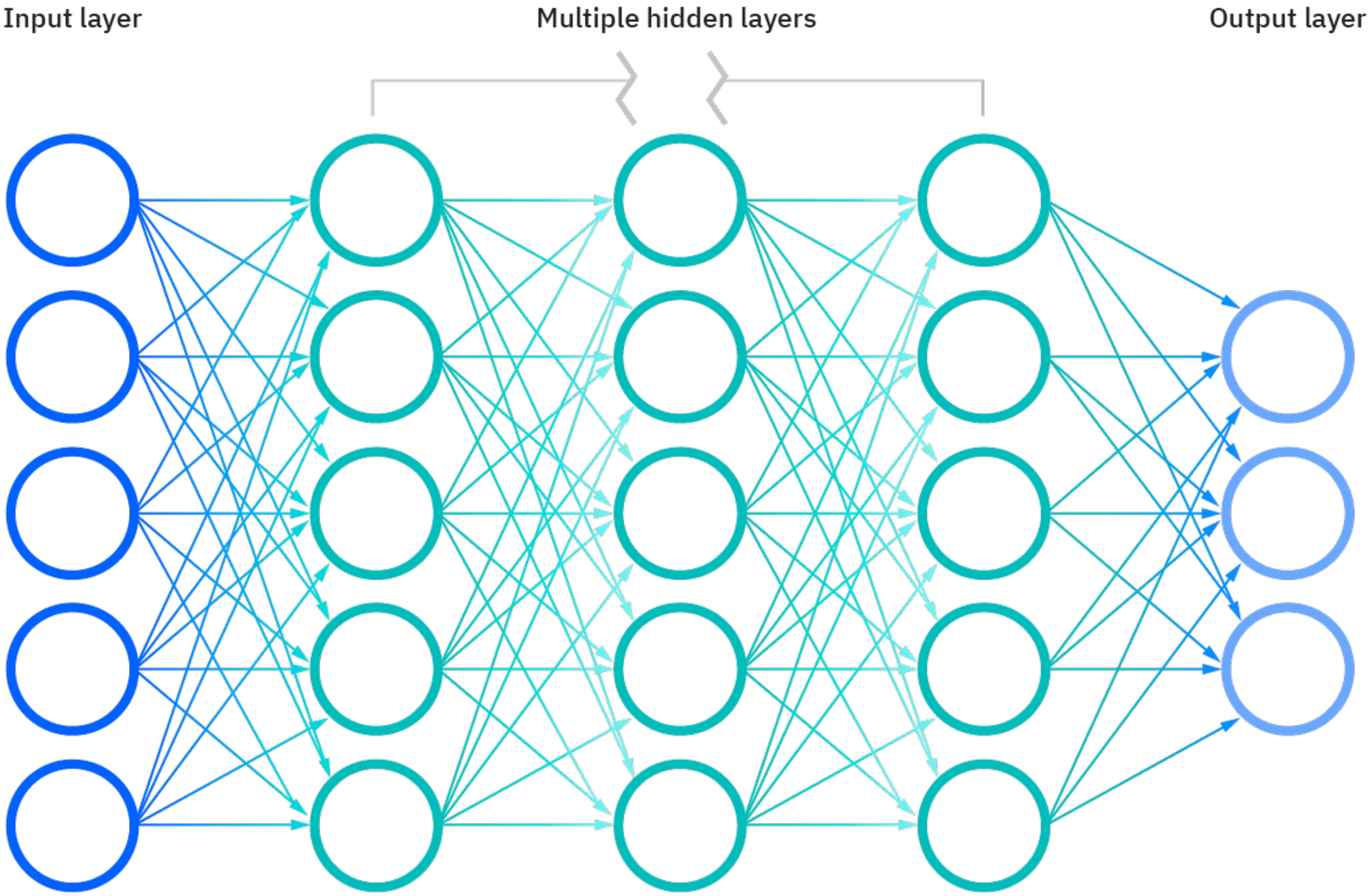
The adjective "deep" in deep learning refers to the use of multiple layers in the network. Early work showed that a linear perceptron cannot be a universal classifier, but that a network with a nonpolynomial activation function with one hidden layer of unbounded width can. Deep learning is a modern variation which is concerned with an unbounded number of layers of bounded size, which permits practical application and optimized implementation, while retaining theoretical universality under mild conditions. In deep learning the layers are also permitted to be heterogeneous and to deviate widely from biologically informed connectionist models, for the sake of efficiency, trainability and understandability, whence the "structured" part.

Refer to wiki

Deep Learning is a Cat

- teach how to move, it learns how to move
- need to prepare food (training data)
- need to tell it how to eat (network structure)
- need to check if it learnt the correct way (evaluation)
- Some accidents will be happened (overfitting, no-coverage, model-collapse, etc)

Deep neural network



Simple example

Training data:

input: 1 output: 2

input: 2 output: 3

input: 3 output: 4

input: 4 output: 5

input: 5 output: 6

Network:

~~*w*(input) + *b*~~

Common structure

Fully connection layer

Convolution layer

RNN layer

…

They will find a good parameters to fit data

The network Learned:

w = 1

b = 1

Practice

Follow ml_examples.py

1. Install pytorch in your machine (cuda version is better, also fine if no cuda)
2. Make sure you can run ml_example.py by next tutorial
3. Check the practices in ml_example.py

More examples: https://github.com/pytorch/examples