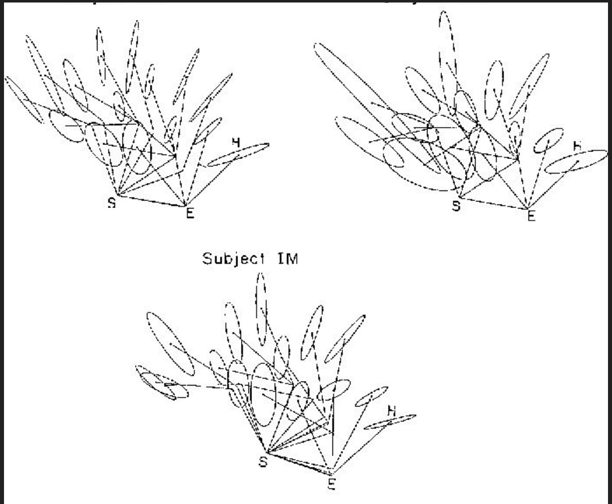
Trajectory Based Human Pose Estimation Using Convolutional Neural Network

Matt McDermott

Introduction

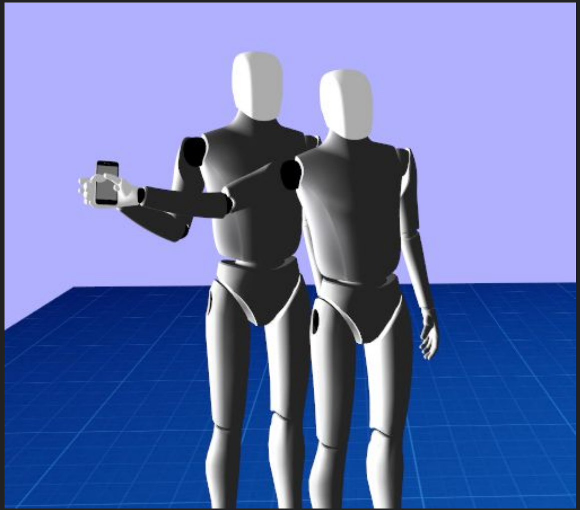
- Multibody systems have configuration dependant inertial properties
  - Observing the force-motion relationship of a system can provide insight on its state
- Kinematic constraints limit some types of motion

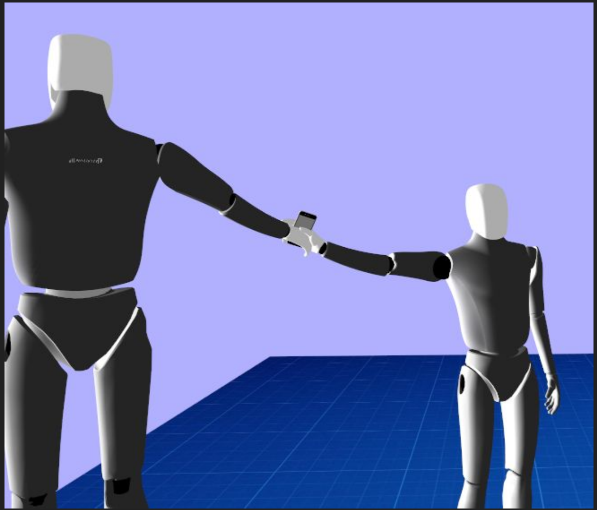


Pictured above: planar case of state dependant endpoint inertia.
Becomes much more complicated in 3D

Goal: Estimate the pose of a human by looking at the trajectory of an object in their hand

- Human arm endpoint impedance is highly correlated with pose
- There are an infinite number of ways to hold an object in one place, however, once you start moving you leave a fingerprint of your inertia & constraints





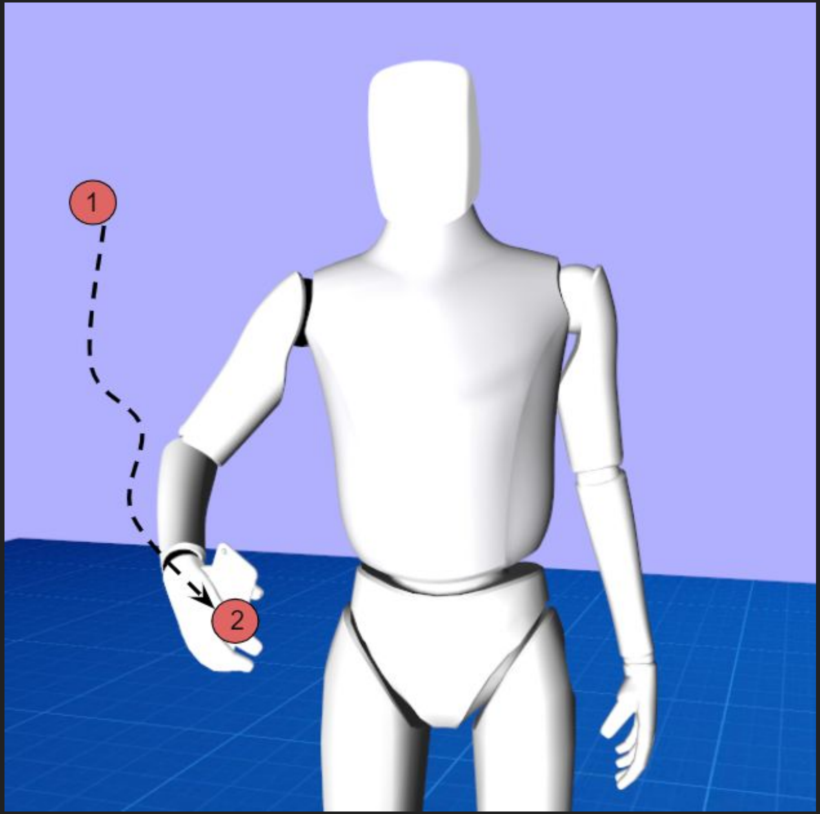
Related Work

- Flash, T., Mussa-Ivaldi, F. Human arm stiffness characteristics during the maintenance of posture. *Exp Brain Res* 82, 315–326 (1990). https://doi.org/10.1007/BF00231251
    - Describes state dependant inertia characteristics of human
- D. Roetenberg et al., “Ambulatory position and orientation tracking fusing magnetic and inertial sensing,” IEEE Trans. Biomed. Eng., vol. 54, no. 5, pp. 883–890, May 2007.
    - Describes sensor drift in accelerometer data: proves feasibility of accelerometer integration for short timesamples
- El-Gohary, Mahmoud, and James McNames. “Human Joint Angle Estimation with Inertial Sensors and Validation with A Robot Arm - IEEE Journals & Magazine.” *IEEE*, IEEE, 12 Feb. 2015, ieeexplore.ieee.org/abstract/document/7041198
    - Approaches a similar problem using a Kalman filter on a robotic arm with KNOWN DYNAMIC CHARACTERISTICS (these parameters can not be easily measured directly on a human)
    - Average 3% error per joint

Assumptions

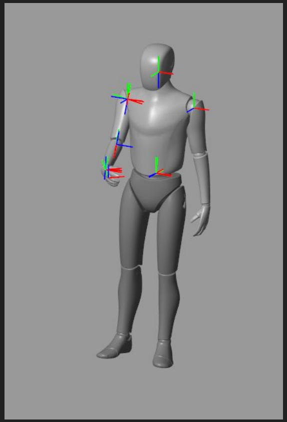
- 9DOF sufficient to model human movement
- Human can compensate for force of gravity
- Human arm follows minimum work trajectory
    - Movement pattern from point A to point B that results from exerting force on hand (with gravity neglected) will result in the same pattern trajectory as if the human consciously exerted the same force to move from A to B
- Measurement device has accurate information on translation and rotation at any time step relative to the start of a trajectory *Rosenberg et al.*
    - Difficult with accelerometers, very easy with AR tags, oculus sensors, robot end effector, etc.

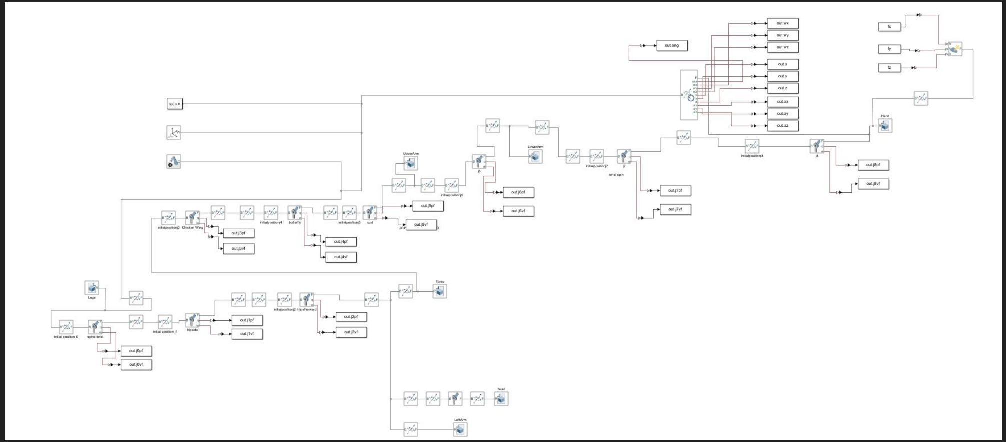
Minimum Work Trajectory



Environment Creation

- Modeled “ragdoll” human in SimScape MultiBody
  - Estimated inertial and damping properties of human body [__]
- 9DOF
  - 3DOF hips, 3DOF shoulder, 1DOF elbow, 2DOF wrist
  - Hips fixed in place relative to world frame
- Fixed simulation step size
- Input- arbitrary forces of hand
- Output- hand movement



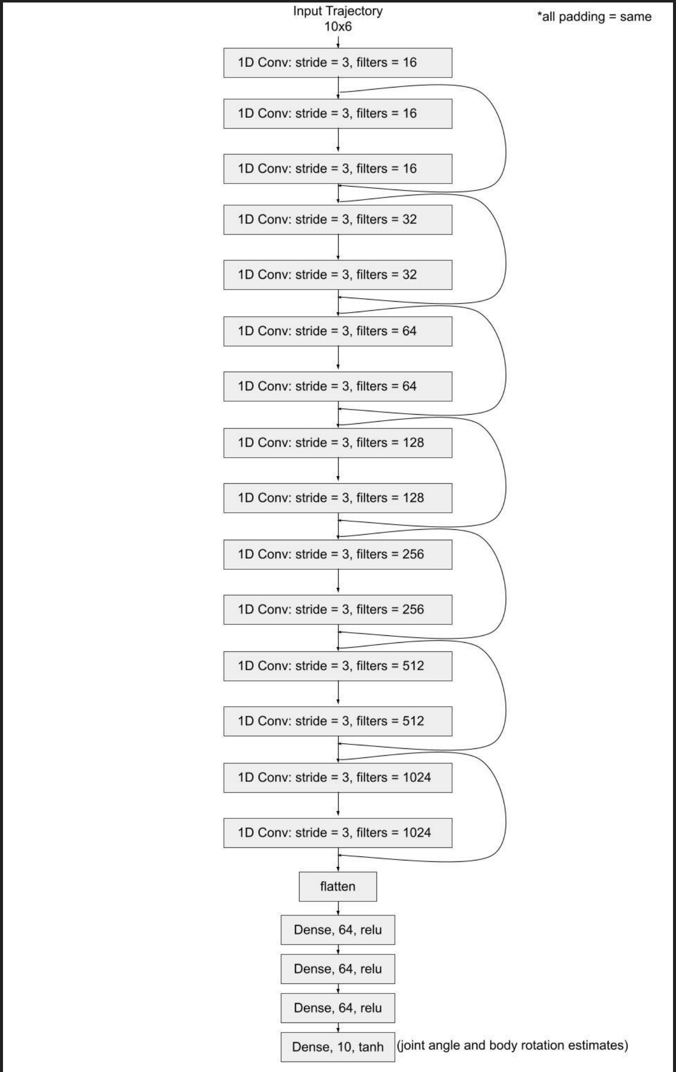


Data Generation

- Create training data consisting of trajectories containing 10 points of information on translation and rotation over the course of 1 second.
  - Simulated with ode3 solver with constant step size of 0.0001s
- Set human model to random initial pose
- Exert a constant linear force of random magnitude and direction for each trajectory
- Save trajectory as the x data and the final pose of the human as the y data
  - Trajectory = translation and rotation in world frame relative to starting position
    - Should work even if you are holding the sensor upside down
- Repeat 1,000,000 times

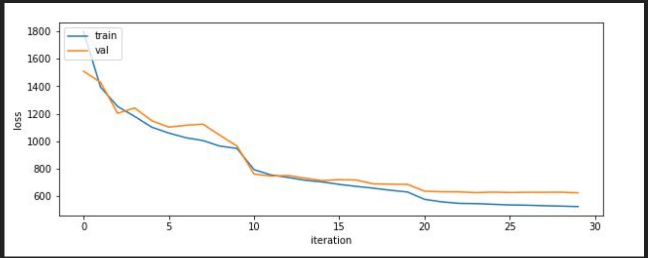
Data Augmentation

- Up to this point, generated data assumes that the human is standing in a constant rotation relative to the world frame
- Ideally our model should not only predict joint angles but also the rotation of the human's hips in the world frame
- Using rotation to cheat and artificially create more data actually hurt performance of model as it promoted memorizing certain patterns rather than learning the underlying dynamics of the system

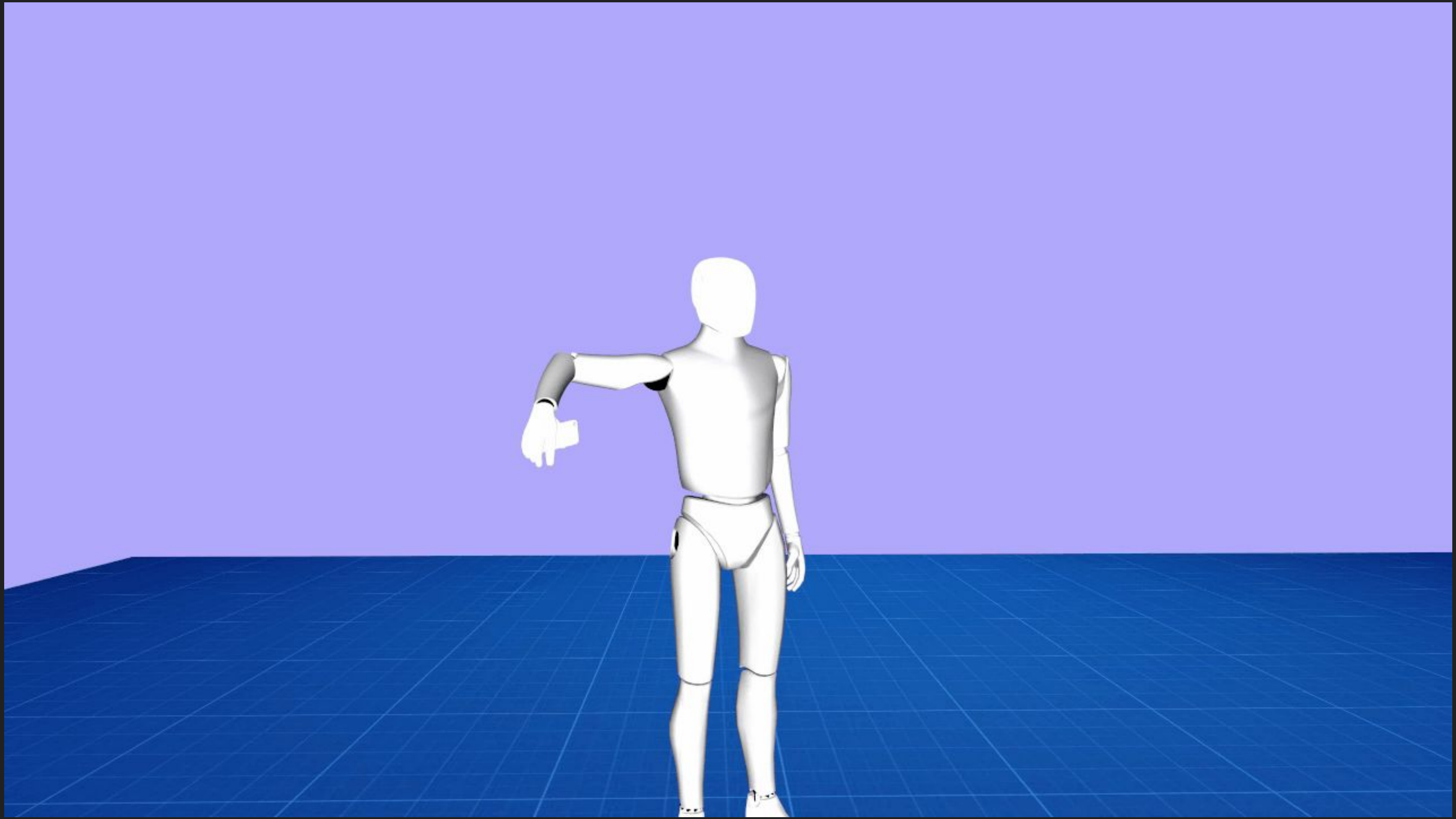

Input Trajectory
10x6
*all padding = same
1D Conv: stride = 3, filters = 16
1D Conv: stride = 3, filters = 16
1D Conv: stride = 3, filters = 16
1D Conv: stride = 3, filters = 32
1D Conv: stride = 3, filters = 32
1D Conv: stride = 3, filters = 64
1D Conv: stride = 3, filters = 64
1D Conv: stride = 3, filters = 128
1D Conv: stride = 3, filters = 128
1D Conv: stride = 3, filters = 256
1D Conv: stride = 3, filters = 256
1D Conv: stride = 3, filters = 512
1D Conv: stride = 3, filters = 512
1D Conv: stride = 3, filters = 1024
1D Conv: stride = 3, filters = 1024
flatten
Dense, 64, relu
Dense, 64, relu
Dense, 64, relu
Dense, 10, tanh
(joint angle and body rotation estimates)

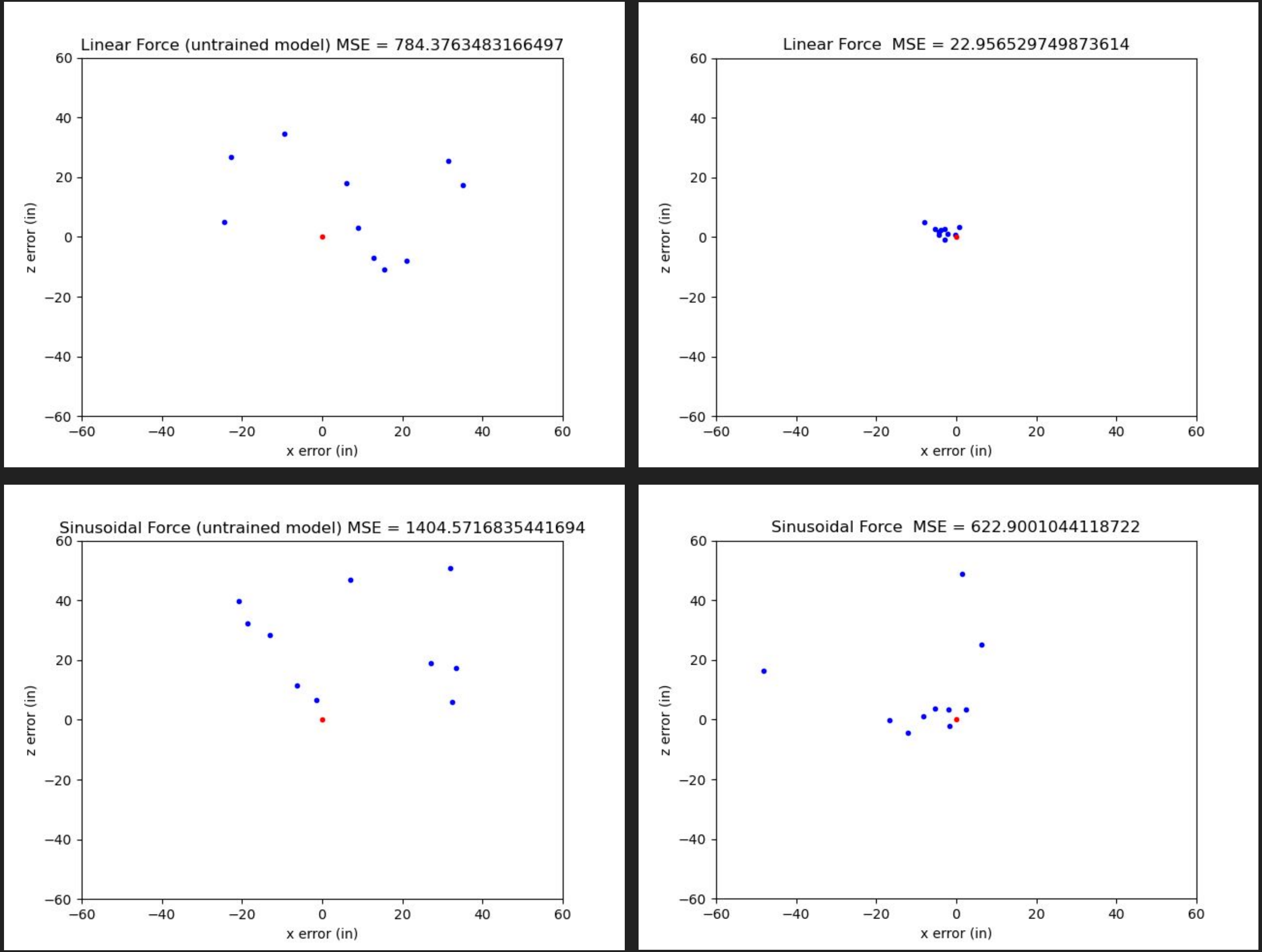
Network Training

- 10 hours on GeForce 1060
- Achieved a best MSE of 371
  - means the average joint estimate was ~19 deg off
  - MSE not a great metric because different joints have different effective ranges (elbow has a range of ~90 deg while shoulder has a range of > 180 deg)
- Average error of 8% per joint
- Body Rotation Problem
  - When the human is rotated close to 180 deg, an estimate of -179 deg will be punished heavily despite being very close to the true value



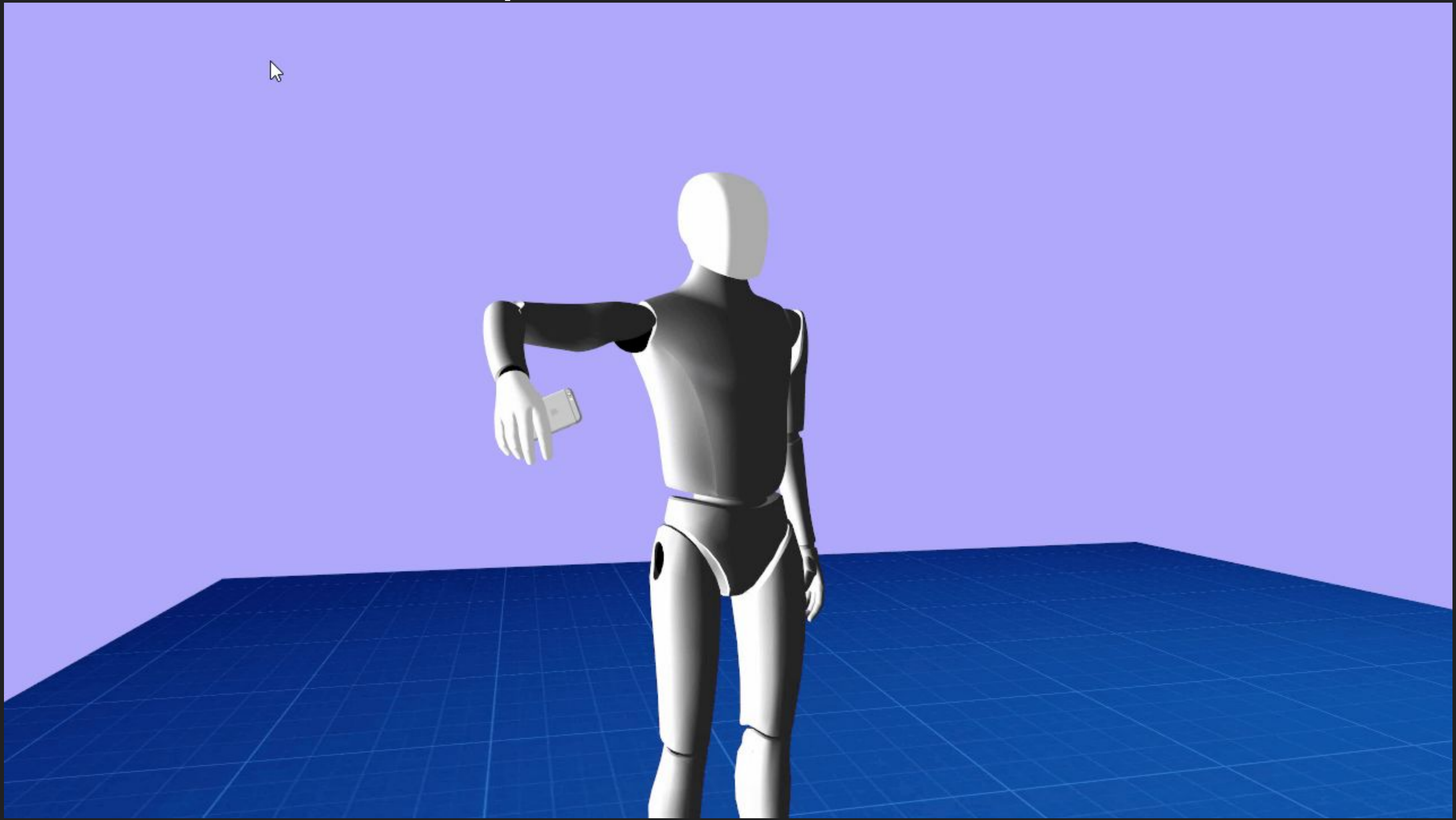
Control: Random Guessing



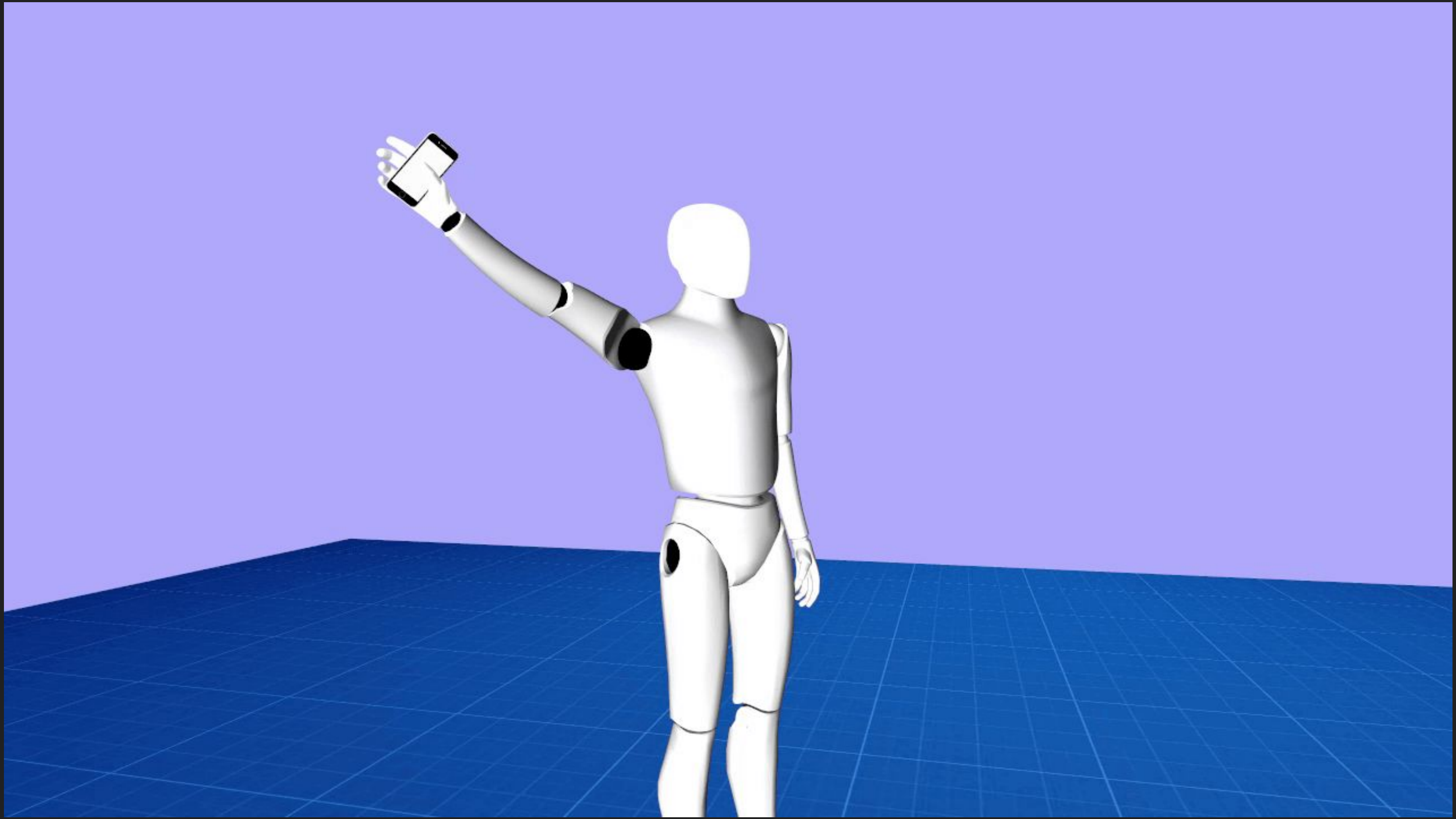


Linear Force (untrained model) MSE = 784.3763483166497
Linear Force MSE = 22.956529749873614
Sinusoidal Force (untrained model) MSE = 1404.5716835441694
Sinusoidal Force MSE = 622.9001044118722
z error (in)
x error (in)
60
40
20
0
−20
−40
−60

Linear Endpoint Force Pose Estimation



Sinusoidal Endpoint Force Pose Estimation



Next Steps

- Modify data generation to train network on acceleration and rotational velocity information so that a cell phone accelerometer and gyro can be used to collect data
- Kalman filter/ Smoothing between 1s samples (right now everything is separate)
- More data
- Improve model of human arm strength characteristics
- Look at real human motion data
- Train on data with nonzero initial endpoint velocity?

Conclusion

- Significant improvement in MSE when guessing poses
- Reasonable performance on randomly varying sinusoidal forces despite being trained on constant forces
    - Suggests this technique may be extrapolated to more general human movement
- Potential real world applications:
    - Accurately representing elbows are in VR
    - High bandwidth pose estimation in powered exoskeletons
    - Provide insight to cobots without additional hardware
        - If a robot knows it is being guided from one direction it can adapt future movements accordingly

References

1. K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.
2. Flash, T., Mussa-Ivaldi, F. Human arm stiffness characteristics during the maintenance of posture. *Exp Brain Res* 82, 315–326 (1990). https://doi.org/10.1007/BF00231251
3. http://www.mbfys.ru.nl/~stan/gielen-MotorControl-2004.pdf
4. El-Gohary, Mahmoud, and James McNames. “Human Joint Angle Estimation with Inertial Sensors and Validation with A Robot Arm - IEEE Journals & Magazine.” *IEEE*, IEEE, 12 Feb. 2015, ieeexplore.ieee.org/abstract/document/7041198
5. http://visal.cs.cityu.edu.hk/static/pubs/conf/accv14-3dposecnn.pdf
6. D. Roetenberg et al., “Ambulatory position and orientation tracking fusing magnetic and inertial sensing,” IEEE Trans. Biomed. Eng., vol. 54, no. 5, pp. 883–890, May 2007.

7. Lin, Feng, and Robert D Brandt. “An Optimal Control Approach to Robust Control of Robot Manipulators.” *Https://Ieeexplore-Ieee-Org.ezproxy.library.tufts.edu/Stamp/Stamp.jsp?Tp=&Arnumber=660845*, IEEE, 1 Feb. 1998.
8. Scott T. Albert, Alkis M. Hadjiosif, Jihoon Jang, Andrew J. Zimnik, Demetris S. Soteropoulos, Stuart N. Baker, Mark M. Churchland, John W. Krakauer, Reza Shadmehr. *Postural control of arm and fingers through integration of movement commands. eLife* (February 11, 2020). doi: https://doi.org/10.7554/eLife.52507.