Trajectory Based Human Pose Estimation Using Convolutional Neural Network

Matt McDermott

May 2021

1 Abstract

Multibody systems have configuration dependent impedance properties due to kinematic constraints and complex inertial relationships between connected link segments. The path of a point fixed to such a system as it reacts to an internal or external force will therefore be influenced by the unique dynamics associated with the current state of the system. If system dynamics can be modeled with sufficient accuracy, it is therefore possible to work backwards and estimate the state of a system by merely observing an endpoint trajectory which can be used to estimate force-motion relationships. Using a Convolutional Neural Network (CNN), this technique can be applied to movement of the human body in order to estimate the pose of an entire human's body by only looking at the movement of an object in their hand.

2 Introduction

Using the fingerprint of virtual endpoint inertia to estimate human pose has numerous advantages over existing human pose estimation techniques. Most significantly, such a technique allows the addition of rudimentary pose estimation into existing systems without the need for additional hardware. For example, using this technique in a VR implementation could provide users with more realistic estimates of elbow configuration without the use of cameras. Furthermore, such a technique has the potential to provide an additional high bandwidth, occlusion-resistant stream of information in settings where there is constant contact between a human and robot. For example, this could be used to improve accuracy in powered exoskeleton devices or to assist in path planning for cobots in collaborative manufacturing operations.

The Minimum Work Trajectory assumption states that human hand movement follows the path of least work when traveling between two points in space. Because of the complex inertial and kinematic relationships between joint segments, these paths are rarely straight lines.

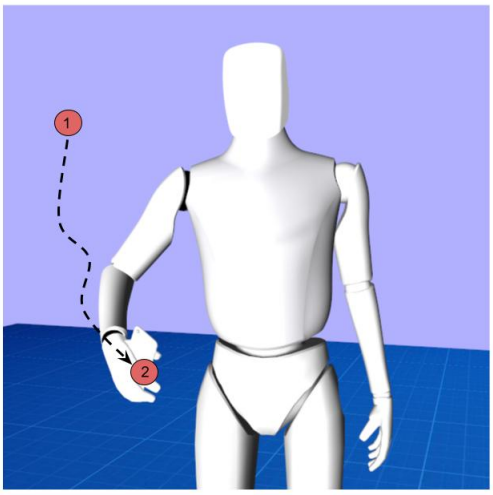

Figure 1: Example Minimum Work Trajectory

If the assumption of the human hand following the minimum work trajectory is correct, then exerting a random force on the endpoint to move the hand from a starting position to an arbitrary ending position will result in the same trajectory as if the human consciously decided to move from point 1 to point 2.

3 Related Works

Flash and Mussa-Ivaldi describe the general properties of endpoint impedance in *Human Arm Stiffness Characteristics During the Maintenance of Posture*. Though they do not provide a complete mathematical representation of stiffness characteristics, their work demonstrates that such patterns of impedance exist and are generally true across all humans. This means that a network trained on data from one human can be used to estimate pose from a dataset recorded from a different human.

Ambulatory Position and Orientation Tracking Fusing Magnetic and Inertial Sensing by Rotenberg et al provides metrics for accelerometer drift when estimating human hand position through the use of a unique Kalman Filter. While previous papers describe overall drift characteristics when integrating accelerometer data to obtain translation and rotation estimates, this paper calculates drift metrics for the specific case in which the recording device is held in the hand of a human. Results from this study indicate that drift in estimates for trajectories of 60 seconds in length will be 10%. Furthermore, the results indicated that drift increases linearly within this time frame. Interpolating these

values would mean that the accuracy of translation estimates obtained from data recorded by an accelerometer held in a human hand for 1 second should have less than 1% error.

Human Joint Angle Estimation with Inertial Sensors and Validation with a Robotic Arm attempts to use accelerometer data from a device fixed to the end effector of a 6DOF robotic arm to estimate the joint angles of the robot. The authors use knowledge of the initial state and dynamics of the robot to construct an Unscented Kalman Filter to arrive at estimates of joint pose. Using this technique, the authors are able to achieve 3% error in their estimates.

Lee, Yoon and Cho use a 1D CNN to categorize human movement patterns from accelerometer data in *Human Activity Recognition From Accelerometer Data Using Convolutional Neural Network*. Despite being a classification task rather than one of continuous output, the authors prove the utility of a CNN for such a task and provide a simple network structure capable of achieving 90% accuracy on a large dataset of accelerometer movement trajectories.

4 Methodology

Training data was generated from scratch using a custom MatLab Simscape MultiBody environment. A 9 degree of freedom (DOF) ragdoll model of a human was created with 3 DOF hips, a 3DOF shoulder, a 1DOF elbow and a 2DOF wrist. Inertial and damping properties were estimated according to [3]. For the sake of creating a balanced dataset the force of gravity was not included when generating training data. This was done to prevent a disproportionate amount of training data being trajectories that involve the arm falling downward. Furthermore, it is generally understood that human motor control subconsciously accounts for the force of gravity acting on each joint segment when executing point to point movement.

Trajectories generated from the MatLab script were augmented by rotating them about the vertical axis and saving the angle at which the body was rotated as an additional degree of freedom in the final joint position (y value). This was done so that the network would be able to recognize poses in which the human's hips are pointing in any direction in the world frame. This strategy also provided the benefit of artificially increasing the amount of training data at minimum computing cost.

A Temporal Convolutional Network (TCN) was chosen for this task in order to best handle the time series data from the trajectory. Trajectories are fed into the input layer of the network and pass through 15 1D Convolution layers, each with a stride of 3, before a flattening operation and 4 linear layers. An architecture similar to ResNet was implemented, with skip connections occurring between each of the convolutional layers. Skip connections make deep networks easier to train, as they result in layers naturally defaulting to identity operations rather than zero. All layers except the output layer used relu activation functions. The output of the network was a tanh function scaled to fit the range of the 10 degrees of freedom. Loss was calculated as the mean squared error

between estimated joint angles and the ground truth joint angles at the end of each trajectory. MSE loss worked very well for all degrees of freedom except the last, which was the value of rotation of the human's hips in the world frame. In rare situations in which the human was rotated nearly 180 degrees, estimates by the network of -180 degrees would be assigned an error of 3602 despite being very close to the ground truth values. Ultimately, this was not a significant issue when training, though it did result in the validation loss asymptotically approaching a nonzero value during training.

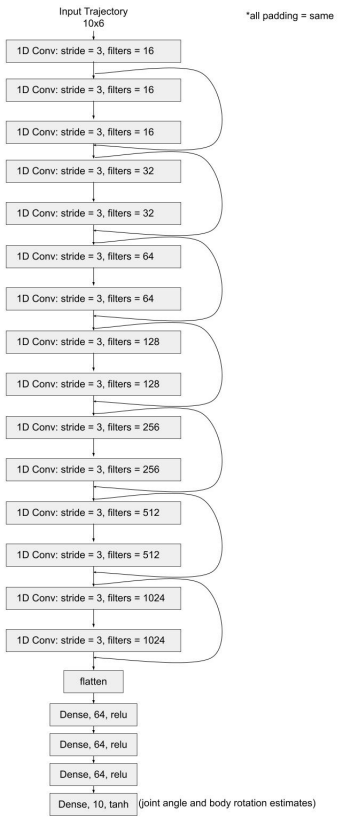

Figure 2: Network Design

5 Experimental Setup and Results

The highest performing network had an MSE loss of 371 which translates to an average error of 8.2% of the possible range of each joint. This value includes a small fraction of estimates in which the loss metric falsely calculated the body rotation to be off by 360°, so the average angle error is likely lower than 8.2%.

An additional test metric was created to analyze the estimated position of the human hips given joint angle estimate outputs from the network. Joint angles from the pose estimates were fed through a forward kinematic model to obtain hip positions on the x-z (horizontal) plane. Hip position error provides a much more useful metric, because it automatically weighs the effects of each degree of freedom differently. Using hip position as a metric for model accuracy also avoids issues with the aforementioned body rotation problem. For example, the MSE weighs shoulder and wrist rotation equally, despite the fact that degrees of freedom near the base of the kinematic chain (such as the hips and shoulder) have a much greater effect on the position of the end effector than those at the end of the kinematic chain.

Before testing the trained network, a control "network" that generated evenly distributed estimates of joint poses at each time step was evaluated on the validation trajectory. These random joint angles were passed through the forward kinematic model to obtain hip position estimates. Passing randomly selected joint angles through the kinematic model rather than merely plotting gaussian distributed points provides a much more accurate baseline to compare the trained network to.

A first validation trajectory was created, consisting of 10 subsequent 1 second trajectory intervals. Each of the 1 second intervals would begin where the previous interval ended, meaning that the ground truth human's hips would be fixed in place for the duration of the validation trajectory. For this first validation dataset, the trajectories were generated in the same way as in the training data; with linear forces of random magnitude moving the human hand from point to point. Both the control and trained networks were given the 10 trajectories from this validation dataset and results were recorded.

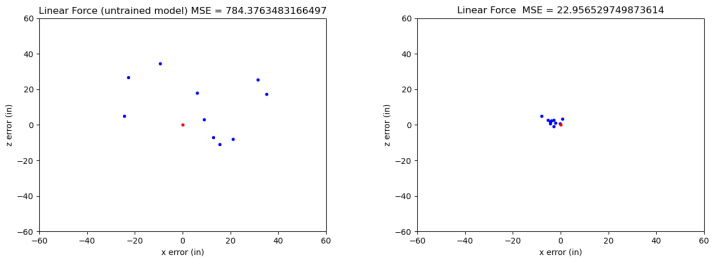

Figure 3: Linear Force Control and Trained Network Performance

An additional validation trajectory was then created, this time using different force characteristics on the hand than were used to train the model. Instead of using linear forces of constant magnitude, this second validation trajectory used sinusoidal forces of randomly varying high frequency, amplitude and phase shift. This resulted in hand movements that would wave back and forth and swing around wildly instead of slow and consistent point to point motion.

It is important to note how the MSE of the control model of the sinusoidal

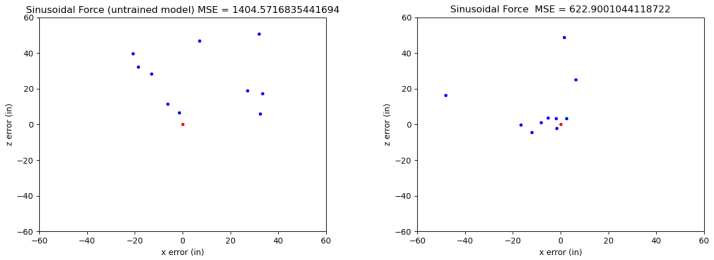

Figure 4: Sinusoidal Force Control and Trained Network Performance

force data is significantly higher than the MSE of the control model on the first linear force dataset. This is due to the fact that the human spent a much larger amount of time with the arm fully extended, meaning that random pose estimations that shared a common endpoint position would on average be further from the hips of the ground truth human.

6 Conclusion

The model achieves high accuracy in pose prediction when applied to data of point to point movement, capable of accurately estimating the location of a user's hips within 5 inches. While this model only achieved an average of 8.2% joint angle error compared to the 3% error of El-Gohary and McNames it should be noted that their technique required knowledge of the initial joint angles as well as accurate knowledge of the dynamic properties of their robotic arm.

Additionally, despite being trained solely on patterns of constant linear force, the final model performed significantly better than random chance on trajectories of noisy sinusoidal forces. If the three outlier points are ignored, the MSE of the sinusoidal force validation dataset drops from 622 to 100 which is significantly better than the MSE of 1400 in the control. Such high performance on a dataset in which the model was not explicitly trained suggests that the model was able to learn some of the underlying characteristics of human arm movement, rather than just learn to match certain trajectory shapes that arise from point to point movement with corresponding joint angles. These results suggest that it may be possible to apply a similar model trained on a simulation generated dataset to a real world human with minimal additional modeling of their mechanical properties.

All training and validation trajectories in this project involved a ground truth human with a static base between subsequent trajectory intervals. Further work could expand this strategy to estimating the pose of a walking or running human. A Kalman filter or additional RNN could be trained to take in sequences of estimates from the network created in this project to obtain the pose of a moving human. Even in the case of a static human, such a technique could be used to smooth out outliers when reading noisy data.

7 References

1. Song-Mi Lee, Sang Min Yoon and Heeryon Cho, "Human activity recognition from accelerometer data using Convolutional Neural Network," 2017 IEEE International Conference on Big Data and Smart Computing (BigComp), 2017, pp. 131-134, doi: 10.1109/BIGCOMP.2017.7881728.

2. 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.

3. 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

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. 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.

6. 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/Stam IEEE, 1 Feb. 1998.

7. 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.

8. Admiraal Marjan, Martijn Kuster, Stan Gielen.. Modeling Kinematics and Dynamics of Human Arm Movement. Motor Control. 2004. http://www.mbfys.ru.nl/ stan/gielen-MotorControl-2004.pdf