# Wearable Upper-limb Exoskeleton and Humanoid Trajectory Planning for Gaze-based Assistance*

Yefan Zhou[1] Sarah Seko[1] Robert Peter Matthew[1] Ruzena Bajcsy[1].

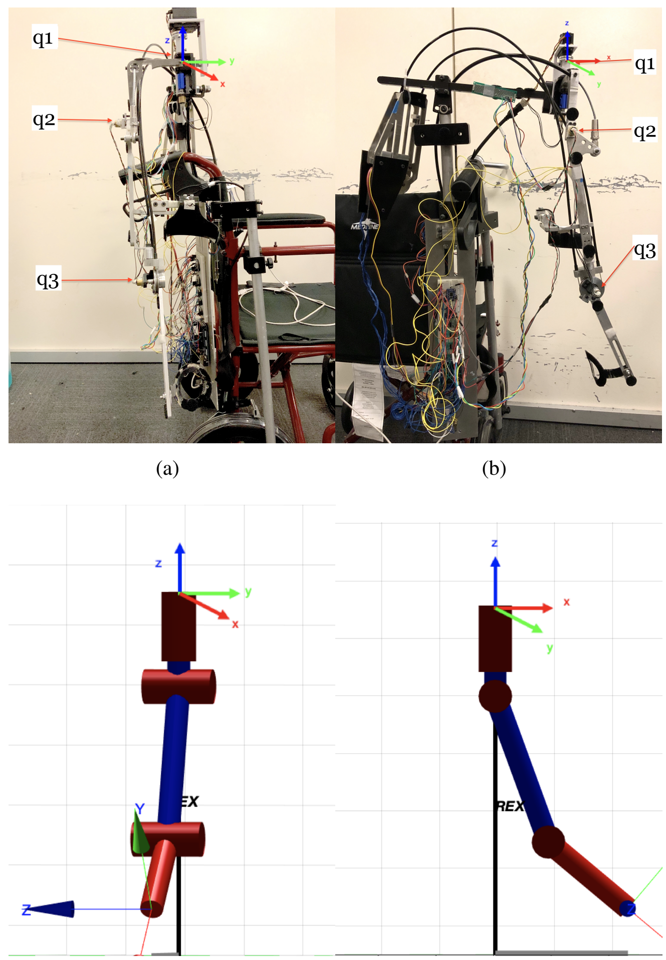
***Abstract*— The integration of eye gaze system and upper-limb exoskeleton has the potential to motivate a new generation of active assistive device. In this work, we propose a novel and affordable active upper-limb exoskeleton development system focusing on system design and trajectory planning algorithm for eye gaze study. We demonstrate its ability to do position and velocity control both in Cartesian space and Joint space, and show that trajectory planning algorithm can be implemented and validated in this platform. In addition, an intuitive and interactive Graphic User Interface(GUI) are designed for the system with functions of 3D visualization and real-time data analysis. Furthermore, we propose two trajectory planning algorithms that are able to generate motion primitive mimicking human movement for robot assistance and validate its performance by motion capture camera system and human subject trial test. The videos are provided here.**

## I. Introduction

Upper-limb exoskeletons are currently explored to assist people suffering from neuromuscular diseases or serve as power augmentation device. The classic exoskeleton design focused on passive assistance. [1] [2] successfully propose a mature method for pneumatic driven exoskeleton to passively provide continuous assistance. With introduction of advanced cognition and vision system with capacity of sensing human intention on certain position in real world, exoskeleton assistance is expected to be more active in order to be integrated to motivate a new generation of rehabilitation or power augmentation device. [3] indicates the high potential of gaze-BCI-driven active robotic assistance. [4] proposes the kinematic design and development of a 6-DOF upper limb exoskeleton for a brain-machine interface (BMI) study. Clearly the exoskeleton control should be not only compliant with user's intention, but should be capable of assisting the movement along human-like generated trajectories. Human arm trajectory mathematic model was first proposed in [5], then developed in [6] [7] [8], which contributed to planning and control development in active assistance [9] [10]. In this work, we propose a 3-DOF upper limb exoskeleton powered by cable and servo motor for eye gaze system study [11]. Based on the platform, trajectory planning optimized by minimum angle jerk criterion and minimum hand jerk criterion and proportional controller designed for joint angle position and velocity control are implemented. In addition, the high accuracy of position control and humanoid smooth trajectory planning are validated in motion capture camera system. Finally, the whole system is integrated by a GUI to achieve interactive and intuitive research development and usage.

## II. System Description

### A. Hardware Design




(a) (b)

(c) (d)

Fig. 1: (a),(c) shows the front view of exoskeleton arm hardware and 3D model. (b),(d) shows the side view of hardware and 3D model. The RGB coordinate arrows marks exoskeleton coordinate system

As shown in Fig.1, The 3-DOF upper-limb exoskeleton has two joints, shoulder and elbow, three generalized coordinates

* This work is a summer project at UC Berkeley Human-Assistive Robotics Technologies Lab (HART), 2019. Please watch our project video link here.

[1] Yefan Zhou is a summer research intern at HART Lab supervised by Prof. Bajcsy. yefan0726@berkeley.edu, bajcsy@eecs.berkeley.edu

TABLE I: D-H Parameter For the Upper-limb Exoskeleton

| Link | $a_i$ | $\alpha_i$ | $d_i$ | $\theta_i$ |
|---|---|---|---|---|
| 1 | 0 | $\frac{\pi}{2}$ | $0.1m$ | $\theta_1$ |
| 2 | $0.27m$ | 0 | 0 | $\theta_2$ |
| 3 | $0.18m$ | 0 | 0 | $\theta_3$ |

TABLE II: Joint Range and Max Angular Velocity

| Joint | | Range | Max speed |
|---|---|---|---|
| Shoulder | $q_1$ | $(-70°, 60°)$ | $5°/s$ |
| | $q_2$ | $(10°, 70°)$ | $1.3°/s$ |
| Forearm | $q_3$ | $(0°, 110°)$ | $3.6°/s$ |

$[q_1, q_2, q_3]$ (in another term, link $[L_1, L_2, L_3]$), representing the shoulder rotation, shoulder elevation and elbow flexion-extension respectively. The D-H Table are shown in Table I. The RGB coordinate axis marker is the defined exoskeleton coordinate(world coordinate). The $q_1$ coordinate is a rotary system consisting metal shaft, rotary encoder, servo motor connected by 3D print hub, driven by servo motor. The $q_2$ coordinate is a lifting system driven by metal cable, which is attached to linear actuator mounted backend. The $q_3$ coordinate is a shaft drive system powered by linear actuator. Angle of each joint is measured by rotary encoder mounted to each link. Considering the human joint limit and trade off between available actuator speed and torque, the rotation range and max angular velocity of each link are set as shown in the Table II. The three links are integrated to move both independently and simultaneously as a system controlled by embedded board Arduino Mega with running frequency 20 $Hz$.

The wearable part consists of two semicircle cuff that used to assist human at upper arm and the forearm. In addition, three load cells are mounted in different interior regions of upper arm cuff to measure the interactive force between human subject and exoskeleton experimentally. A similar force sensor system is expected to mount on forearm cuff for future development.

### B. Software Design

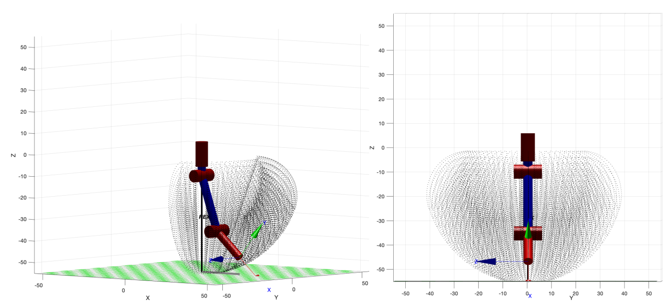
The software system of exoskeleton includes a Graphic User Interface, trajectory planner, low-level controller, sensor filter. Inspired by Rviz[10] and based on our final goal to be integrated with high level command system, we design a Graphic User Interface in order to set end effector (hand) position setpoint in Cartesian Space either by typing in manually or choosing with cursor in 3D visualization world. This interface is expected to be connected with gaze system to gain setpoint in real time in further development. In addition, the interface provides input box for tunable parameters like trajectory duration and feedback $K_p$, 3D visualization for simulation results and data auto-plot for comparison of hardware performance and simulation results. For detail of the entire operation in GUI, please refer to the video.

Trajectory planner is implemented both in PC side in MATLAB and in embedded board side in C program for each joint respectively, reasons for this scheme is, implementation in MATLAB would help faster computation speed of inverse kinematic and curve fit, which would be mentioned in next section while implementation in embedded board would contribute to smooth trajectory with high planning update frequency. Then transmit the parameters via USB cable communication and assign the value to variable in C program. In embedded end, the planner generates angle trajectory for each joint, and updates position, velocity in discrete time step as the input of low-level controller during execution of control. The execution scheme is based on the assumption that controller is highly responsive and temporary setpoint can be reached in tiny time step. Although our hardware setup is not able to fully satisfy the assumption, the method is validated to generate smooth and accurate hardware trajectory performance in the experiment. In the execution of planner and controller, the joint angle state will be transmitted back as data log for data analysis.

## III. Algorithm

### A. Forward/Inverse Kinematic

Based on our hardware setup, the forward and inverse kinematic in our exoskeleton can be computed purely by geometric analysis and one Cartesian coordinate $(x, y, z)$ maps to one joint state $(q_1, q_2, q_3)$. In Fig.2, we iterate through our Joint-space in one degree unit to construct the Workspace visualization. Since inverse kinematic in this work is determinate and accurate in theory, we adopt joint velocity controller for Cartesian and Joint space trajectory.



(a) (b)

Fig. 2: Workspace of 3D Exoskeleton Model

### B. Trajectory Planning

Referring to the work in [12], reaching motions restricted to an horizontal plane in front of chest is planned at the task-oriented coordinates, which are solely dependent on the kinematics of movement and the hand path mostly is a straight path with a bell-shaped speed profile. As proposed in their work, we choose Minimum Hand Jerk Criterion for this region's movement.

Elevation movements in sagittal plane mostly in one side of body, as being proposed in [7] and [8], are more affected by gravitational force, viscosity of joints and other dynamics factors, and the hand paths are roughly straight or gently curved, so Minimum Torque Change Criterion is commonly considered for this region's movement. However, considering the high time complexity of iterative method to solve the non-linear optimization, we adopt the an approximate algorithm Minimum Angle Jerk Criterion for this introduction work.

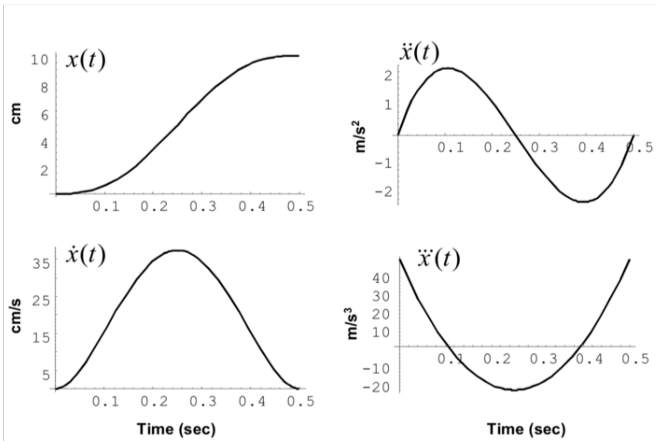



Fig. 3: Trajectory Curve of Minimum Jerk Algorithm. ($x(t)$ is Cartesian space position for Minimum Hand Jerk and joint angle for Minimum Angle Jerk)

- **Minimum Hand Jerk Criterion**: A trajectory is planned according to arm position $(x, y, z)$ in Cartesian space so as to minimize the time integral of the jerk (triple differentiation of the hand position with respect to time):

$$C_{HJ} = \frac{1}{2}\int_0^{t_f}\left\{\left(\frac{d^3x}{dt^3}\right)^2 + \left(\frac{d^3y}{dt^3}\right)^2 + \left(\frac{d^3z}{dt^3}\right)^2\right\}dt \quad (1)$$

A fifth polynomial is chosen in time to represent position for axis variable $(x, y, z)$ respectively.

$$x(t) = a_0 + a_1t + a_2t^2 + a_3t^3 + a_4t^4 + a_5t^5 \quad (2)$$

Any point to point movement optimized by the criterion has the constraint as:

$$x(t) : \begin{cases} x(0) = x^s & x(t_f) = x^f \\ \dot{x}(0) = 0 & \dot{x}(t_f) = 0 \\ \ddot{x}(0) = 0 & \ddot{x}(t_f) = 0 \end{cases} \quad (3)$$

As derived in Flash and Hogan [6], solution to this system can be obtained as:

$$\begin{aligned} x(t) =& x^s + \left(x^s - x^f\right) \\ & \cdot\left\{-10\left(\frac{t}{t_f}\right)^3 + 15\left(\frac{t}{t_f}\right)^4 - 6\left(\frac{t}{t_f}\right)^5\right\} \\ & (0 \leqq t \leqq t_f) \end{aligned} \quad (4)$$

For implementation in MATLAB, we first solve Eq.2 and Eq.3 represented as matrix form, getting trajectory in discrete time step represented as $[(x_0, y_0, z_0)...(x_t, y_t, z_t)]$. Inverse Kinematic is applied to each position point to get corresponding three link angle represented as $[(q_{10}, q_{20}, q_{30})...(q_{1t}, q_{2t}, q_{3t})]$. The results are curves in position, velocity and acceleration for each link, we then use joint angle position represented as fifth polynomial

$$q(t) = a_0 + a_1t + a_2t^2 + a_3t^3 + a_4t^4 + a_5t^5 \quad (10)$$

and its derivation represented by fourth polynomial polynomial to fit the curves, gain the value for coefficient $[a_1, a_2, ..., a_5]$. The coefficients are then assigned to Eq.4 and its derivation, which are implemented in C code in embedded board, the planner will update the $(x_t, v_t)$ as the input of controller in real time. The planning result is a straight-line trajectory in the Cartesian coordinate system with a bell shape of tangential velocity.

- **Minimum Angle Jerk Criterion**: A trajectory based on the minimum angle jerk criterion is planned in Joint Space. The objective is to minimize the integral of the jerk over the motion duration, as follows:

$$C_{AJ} = \frac{1}{2}\int_0^{t_f}\sum_{i=1}^{4}\left(\frac{d^3\theta_i}{dt^3}\right)^2 dt \quad (5)$$

$\theta_i$ expresses the angle of joint coordinate i. The formulation and solution is in the same manner as the minimum hand jerk model. The trajectory is expressed by fifth-order polynomial for joint angle variable $(q_1, q_2, q_3)$ rather than axis variable. For implementation in MATLAB, replacing $x_t$ with $q_t$ and solving the Eq.2 and Eq.3, the results are shown as $[(q_{10}, q_{20}, q_{30})...(q_{1t}, q_{2t}, q_{3t})]$. For implementation in C program, we assign the value of $x_s$, $x_f$, $t_f$ in Eq.4 according to predefined start and end point and tunable parameter time duration of trajectory.

### C. Low-Level Controller

The input of joint angle controller is position and velocity $(x_t, v_t)$ with respect to time. Due to sensor accuracy, in this preliminary work we fail to get reliable measurement of angle velocity (angle reading resolution is degree up to the nearest integer while reading time interval is $50ms$) but only position from rotary encoder. Considering different functions of motor, servo motor only allows position control of rotation angle and linear actuator only allows velocity control, we adopt open-loop position controller in tiny time step for coordinate $q_1$ driven by servo motor(shoulder plane rotation) with higher update frequency from planner. We adopt open-loop velocity controller with position proportional feedback to control in larger time step for $q_2$ and $q_3$ driven by linear actuator(shoulder elevation and elbow flexion-extension) with low update frequency from planner. We failed to increase update frequency for $q_2$ and $q_3$ due to low responsiveness of mechanical system (inadequate torque provided by linear actuator and loaded with arm weight).

### D. Sensor Filtering

The purpose of sensor filtering is to get high resolution reading of rotary encoder angle while maintaining fast responsiveness. We have tested average filter and exponential filter with variant window size and coefficient and finally adopted average filter for sake of lower deviation between filtered reading and ground truth position while exponential filter will sacrifice reading accuracy for higher running rate.

## IV. Experiment and Results

### A. Trajectory Validation

The objective of the experiment is to validate the accuracy of trajectory performed by exoskeleton hardware in real world compared with predefined trajectory in simulation. Since our control and sensor reading is based on Joint space and the trajectory in Cartesian space is inferred by forward kinematic and arm part length, we should have a external sensor system to capture the hardware performance and compare it with our inference(simulation).

The experiment setup is a motion capture camera system comprising of ten depth cameras, and LED markers as is proposed in [13]. We designed eight experiments for different arm elevation and reaching tasks without loading with human subject.

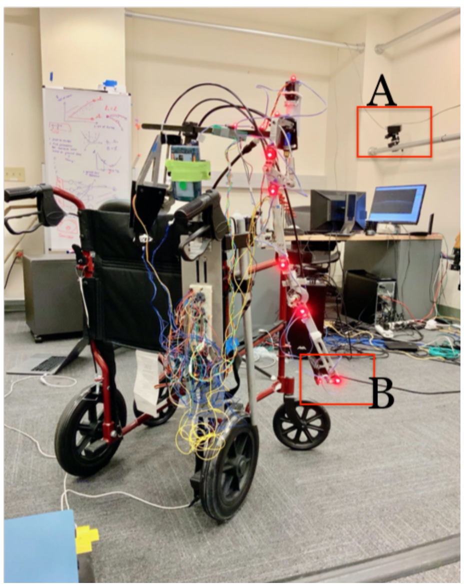
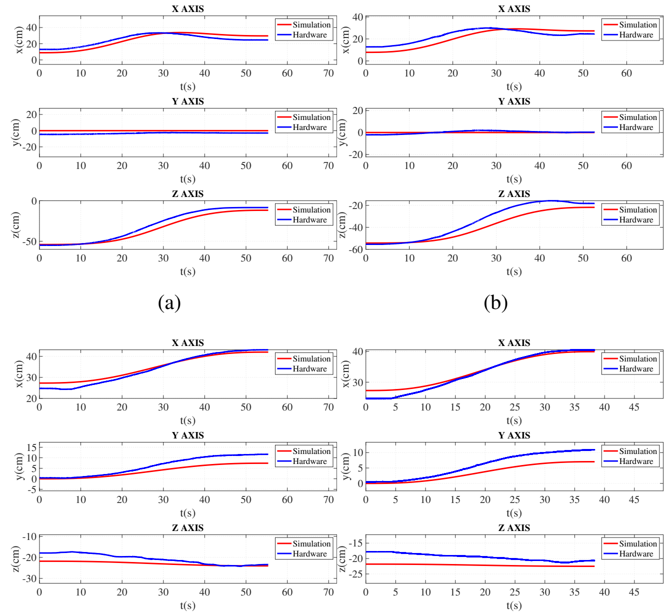



Fig. 4: A is motion capture camera, B is LED markers

Part of results are shown in Fig.5, the $x$ axis is time and $y$ axis is the position of end effector in exoskeleton coordinate, the red line is simulation trajectory while the blue line is hardware trajectory. Fig.5(a) and (b) shows the elevation movement in sagittal plane with minimum angle jerk algorithm, the simulation curve is inferred by forward kinematics. Fig.5.(c) and (d) shows the straight-line movement in nearly horizontal plane with minimum hand jerk algorithm, the simulation curve is generated by algorithm directly. The results indicate that the exoskeleton executed the trajectory planned in Joint Space very well with smooth trajectory and low derivation but the performance of trajectory planned in Cartesian Space is not desirable. On the one hand, the problem lies on the synchronization of three links with different max actuation speed and torque. On the other hand, it indicates that current joint velocity controller is unstable without velocity measurement and feedback. In addition, as proposed in [14], a cable tension controller might be necessary for realizing more accurate torque control for cable-driven system.




(a) (b) (c) (d)

Fig. 5: The Simulation and Hardware Real Trajectory Comparison

### B. Human Subject Trial Test

For the preliminary work, we make unofficial test for healthy human subject to get intuitive assistive experience from the exoskeleton. The feedback indicates the assistive torque on upper arm is too small to lift entirely relaxed arm part and unable to satisfy the velocity bell shape in trajectory planned. The record video is presented here.

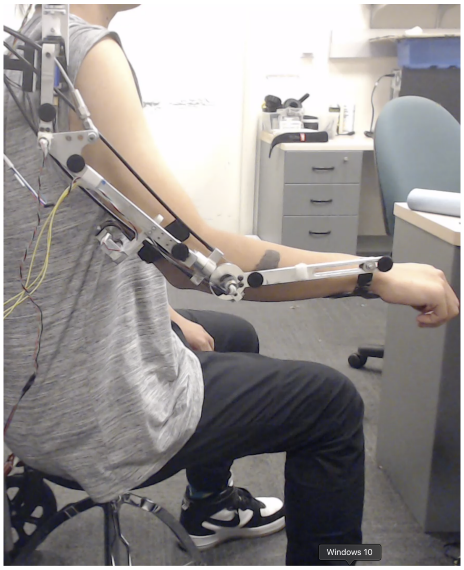


Fig. 6: The Scene of Human Subject Trial Test

## V. Conclusion and future work

In this work, we made an introduction and initial exploration of upper-limb exoskeleton for gaze-based arm assistance. We developed it both in simulation and hardware and demonstrated its capacity of executing humanoid trajectory. Future improvements are proposed as following:

- *System* The mechanical design of exoskeleton is expected to get improvement of increasing applying torque for each joint. The possible method could be using servo and linear actuator with larger torque to replace current setup.
- *Planning* For this work, we manually decide which humanoid trajectory planning to use for certain motion. It would be reasonable to have a more high-level path planning algorithm which can plan a continuous path combining different trajectory planning for different region. Besides, a fast computation method to solve optimization with minimum torque change criterion is expected to develop.
- *Low-Level Controller* The point to be improved of low-level controller is the synchronization of three joints. The method proposed in [15] has the potential to achieve fully synchronization performance. In addition, a cable tension controller might help contribute to more accurate position control.

## References


[1] R. P. Matthew, E. J. Mica, W. Meinhold, J. A. Loeza, M. Tomizuka, and R. Bajcsy. Introduction and initial exploration of an active/passive exoskeleton framework for portable assistance. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5351–5356, Sep. 2015.

[2] D. Kaneishi, R. P. Matthew, and M. Tomizuka. Optimal control parameterization for activelpassive exoskeleton with variable impedance actuator. In *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob)*, pages 713–719, Aug 2018.

[3] A. Frisoli, C. Loconsole, D. Leonardis, F. Banno, M. Barsotti, C. Chisari, and M. Bergamasco. A new gaze-bci-driven control of an upper limb exoskeleton for rehabilitation in real-world tasks. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6):1169–1179, Nov 2012.

[4] Junkai Lu, Wenjie Chen, and Masayoshi Tomizuka. Kinematic design and analysis of a 6-dof upper limb exoskeleton model for a brain-machine interface study. *IFAC Proceedings Volumes*, 46(5):293–300, 2013.

[5] N Hogan. An organizing principle for a class of voluntary movements. *Journal of Neuroscience*, 4(11):2745–2754, 1984.

[6] T Flash and N Hogan. The coordination of arm movements: an experimentally confirmed mathematical model. *Journal of Neuroscience*, 5(7):1688–1703, 1985.

[7] Y. Uno, M. Kawato, and R. Suzuki. Formation and control of optimal trajectory in human multijoint arm movement. *Biological Cybernetics*, 61(2):89–101, Jun 1989.

[8] CG Atkeson and JM Hollerbach. Kinematic features of unrestrained vertical arm movements. *Journal of Neuroscience*, 5(9):2318–2330, 1985.

[9] M. J. Johnson, K. J. Wisneski, J. Anderson, D. Nathan, and R. O. Smith. Development of adler: The activities of daily living exercise robot. In *The First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006. BioRob 2006.*, pages 881–886, Feb 2006.

[10] E. T. Wolbrecht, V. Chan, V. Le, S. C. Cramer, D. J. Reinkensmeyer, and J. E. Bobrow. Real-time computer modeling of weakness following stroke optimizes robotic assistance for movement therapy. In *2007 3rd International IEEE/EMBS Conference on Neural Engineering*, pages 152–158, May 2007.

[11] X. Dong, H. Wang, Z. Chen, and B. E. Shi. Hybrid brain computer interface via bayesian integration of eeg and eye gaze. In *2015 7th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 150–153, April 2015.

[12] W K Abend, Emilio Bizzi, and P. Morasso. Human arm trajectory formation. *Brain : a journal of neurology*, 105 Pt 2:331–48, 1982.

[13] R. P. Matthew, S. Seko, R. Bajcsy, and J. Lotz. Kinematic and kinetic validation of an improved depth camera motion assessment system using rigid bodies. *IEEE Journal of Biomedical and Health Informatics*, 23(4):1784–1793, July 2019.

[14] Y. Mao and S. K. Agrawal. Wearable cable-driven upper arm exoskeleton - motion with transmitted joint force and moment minimization. In *2010 IEEE International Conference on Robotics and Automation*, pages 4334–4339, May 2010.

[15] Antonio Frisoli, Claudio Loconsole, Riccardo Bartalucci, and Massimo Bergamasco. A new bounded jerk on-line trajectory planning for mimicking human movements in robot-aided neurorehabilitation. *Robotics and Autonomous Systems*, 61(4):404–415, 2013.