École Polytechnique Fédérale de Lausanne

X-jaw

Master thesis

Author :

Barbara De Groot, 296815

CREATE Lab, Prof. Josie Hughes
Supervisor : Benhui Dai

June 1, 2025

Contents

1 Introduction

- Jaw is one of the most complex articulation in the human body with its 6 degrees of freedom and complex joint movement
- Chewing robot applications: food industry, medical field, dental field, etc.
- Chewing robot can be used to test food texture, dental implants, orthodontic devices, etc.
- Chewing robot can be used to study chewing disorders and develop treatments / also understand more about the human chewing process/development throughout life
- State of the art chewing robots
- Most of them are not able to mimic the human chewing process because not 6 dof / no saliva / no tongue / not able to apply the same forces as human
- Our aim is to create a chewing robot that can mimic the human chewing process as closely as possible

2 Methods

2.1 Required workspace and forces –> criteria for design

- max chewing force across literature
- max range of motion across literature

2.2 Mechanical design

- goal is to create a robotic jaw that can mimic the motion and force of human chewing
- 6dof stewart platform to be able to mimic the motion of the jaw
- linear actuators instead of rotary servo motors to have more efficient force transmission + simpler kinematics + more rigid structure
- choice of actuators based on the required force to mimic human chewing (speed less important as jaw can chew even if slow) + required length to reach the desired range of motion + feedback to control in position
- choosing the dimensions of the stewart platform based on the size of the actuators + working space of the robot
- choice of structure/material to hold upper jaw to be rigid enough to not deform under the forces applied by the actuators
- 3 axis load cells to measure the force applied by the jaw
- so far 3d printed teeth/jaw but to be changed in the future

2.3 Control

2.3.1 Hardware (electronics)

A Teensy 4.1 (600 MHz ARM Cortex-M7, single-precision FPU) executes the control loop.Its key peripherals are:

- three 12 A dual DC motor drivers (DF Robot) controlling the six linear actuators;
- six analogue inputs reading potentiometer position feedback from the linear actuators;
- three transmitters for the load cells mounted on top of the maxilla;
- an on-board micro-SD slot used for trajectory files and calibration data.

A 12 V AC/DC brick powers the actuators directly; the user's computer supplies 5 V to the Teensy, which in turn sources the 3.3 V logic rails for the motor drivers and load-cell transmitters. The full electronics schematic is shown in Fig. 1.

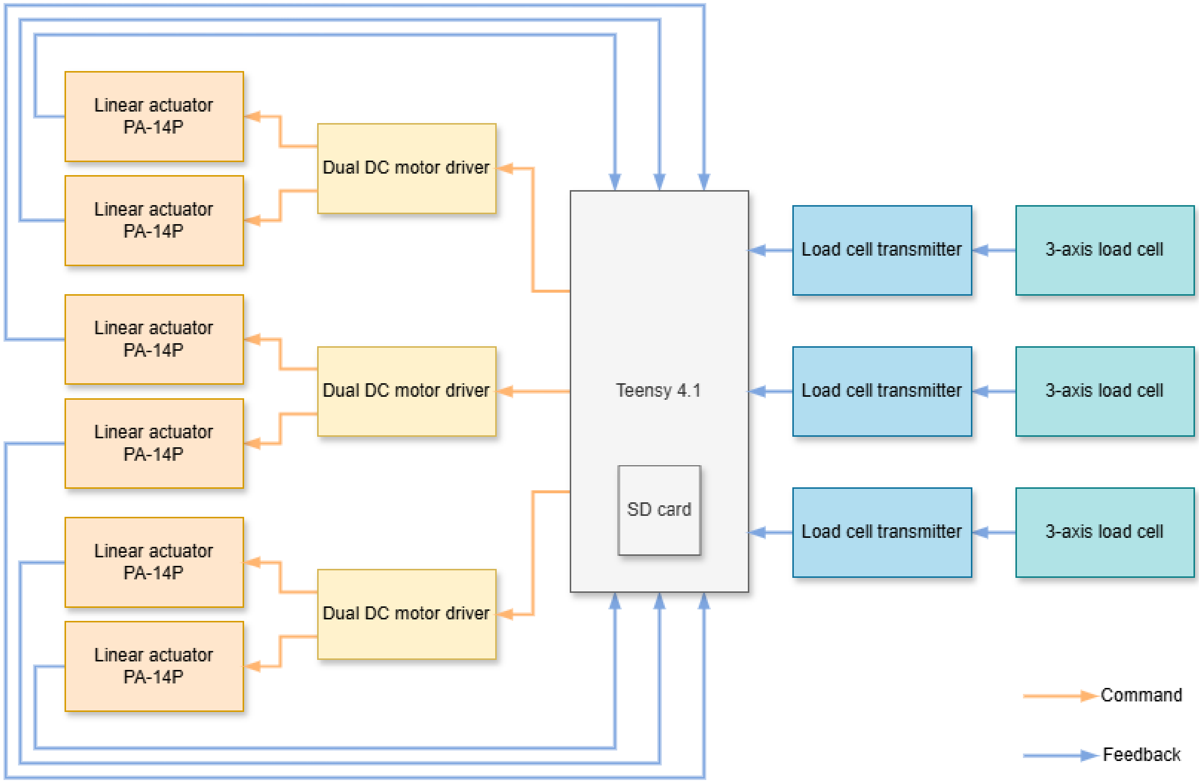

Figure 1: Electronics schematic.

2.3.2 Software architecture

The central class `RobotController` maintains the finite-state machine in Fig. 3 and manages two modules:

- `StewartPlatform`: inverse kinematics, trajectory interpolation, and low-level actuator commands;
- `ForceSensing`: continuous load-cell acquisition and filtering;

The controller is designed to be modular, allowing for easy addition of new modules such as a tongue or saliva module in the future. The three controller states are:

1. **Stop** – return to home pose; reload trajectory if the user selects a new file;
2. **Calibrate** – user can manually change the initial (x, y, z) position via the GUI;
3. **Move** – replay the selected trajectory.

A lightweight Python GUI on the host PC issues high-level commands, such as state changes, and plots sensor data.

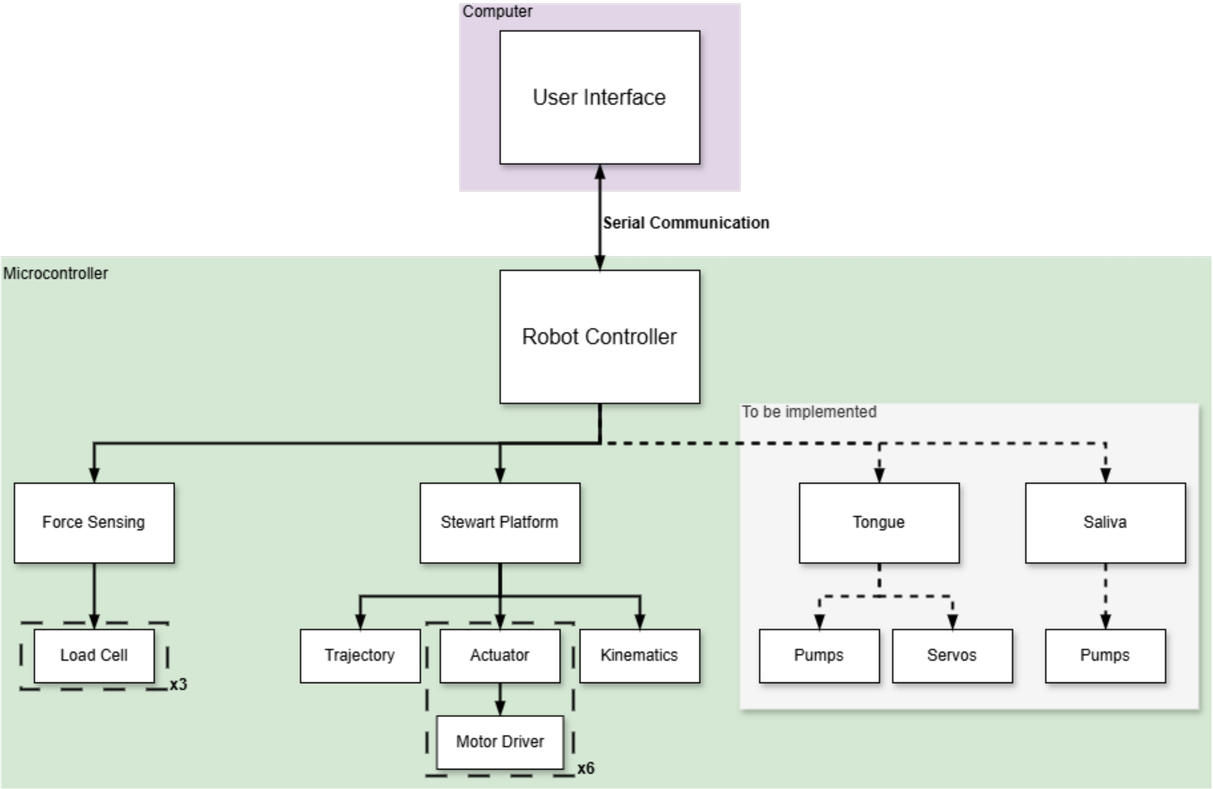

Figure 2: Overall code structure.

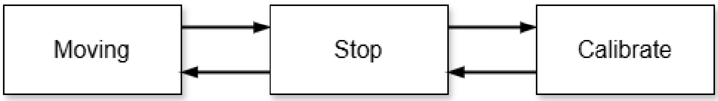

Figure 3: Robot controller state machine.

2.3.3 Position control

The Stewart Platform follows a 3D trajectory (x, y, z, roll, pitch, yaw) from a .csv file on the micro-SD card. See section 2.4 for details on the recording protocol and data processing. Each pose is defined by its position $\mathbf{t} = (x, y, z)$ and orientation given by the Euler angles (ϕ, θ, ψ), which are the roll, pitch, and yaw angles respectively. The trajectory is then linearly interpolated with a fixed time step chosen by the user.

Inverse kinematics For each pose in the trajectory, `Kinematics` computes the lengths of the six linear actuators that will achieve the desired pose of the platform, i.e. the inverse kinematics.

To do so, we first compute the standard rotation matrix $R(\phi,\theta,\psi)$ for the Euler angles, which is defined as the product of three rotation matrices about the Z, Y, and X axes:

$$R(\phi,\theta,\psi) = R_Z(\psi)R_Y(\theta)R_X(\phi) = \begin{pmatrix}\cos\psi & -\sin\psi & 0\\ \sin\psi & \cos\psi & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}\cos\theta & 0 & \sin\theta\\ 0 & 1 & 0\\ -\sin\theta & 0 & \cos\theta\end{pmatrix}\begin{pmatrix}1 & 0 & 0\\ 0 & \cos\phi & -\sin\phi\\ 0 & \sin\phi & \cos\phi\end{pmatrix}$$

The platform joints $\mathbf{p}_i$, i being the actuator index, are then rotated about a fixed point $\mathbf{c}$, which is the front of the gnathion, and translated by the user-defined home position $\mathbf{t} = (x,y,z)$, resulting in the world coordinates of the platform joints:

$$\mathbf{w}_i = R(\mathbf{p}_i - \mathbf{c}) + \mathbf{c} + \mathbf{t}.$$

Finally, the actuator length is the Euclidean distance to the fixed base joint $\mathbf{b}_i$:

$$\ell_i = \|\mathbf{w}_i - \mathbf{b}_i\|_2.$$

PI controller For each actuator, its desired length is sent to a PI position controller, see Figure 4. To minimize the noise of the potentiometer feedback, we apply a low-pass filter averaging the last 10 samples.

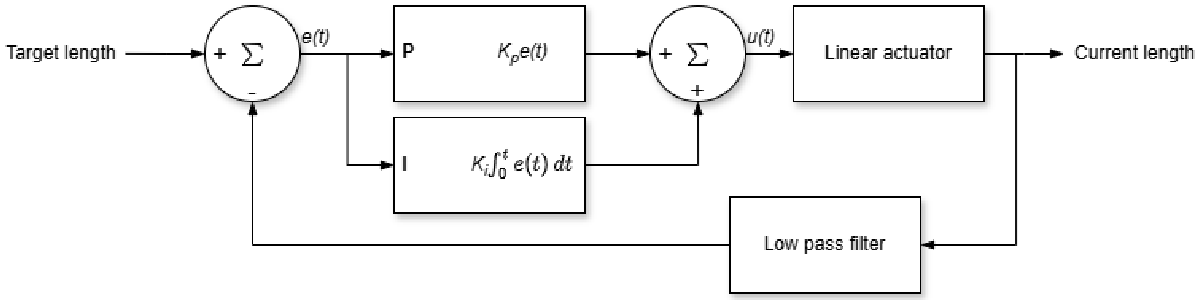

Figure 4: Position PI controller for the linear actuators.

2.4 Data acquisition and processing

Subjects. Two healthy adult volunteers (author and project supervisor) participated in this pilot recording. Owing to time constraints and the exploratory nature of the study, no additional subjects were recruited.

Motion-capture acquisition. Mandibular motion was recorded with a five-camera OptiTrack system sampling at 120 Hz. Four reflective markers arranged in a square were attached to the forehead and served as a head-fixed reference frame. A second set of three markers forming a triangle was placed on the gnathion. Two additional lip markers were recorded but later discarded because a single marker cannot encode orientation. [1, 2]

The subject then performed the motion sequences listed in Table 1. Each frame was saved by Motive as a `.csv` file that contains the 3-D marker positions (in millimetres) and the orientation of each marker set as quaternions. The calibrated volume had a residual error of 0.3 mm.

Food	**Motion**	***Optional:*** **Duration**
Empty mouth	20× open–close cycles	—
Chewing gum (Xylit-Pro, *Excitemint*)	Random side chewing	2 min
	Right-side chewing	1 min
	Left-side chewing	1 min
	Front-teeth-only chewing	1 min
Biscuits (Bretzeli, *Kambli*)	random chewing	—
	front-teeth chew → right-side chew	—
	front-teeth chew → left-side chew	—
	fast random chewing	—
	slow random chewing	—

Table 1: Recording protocol. *Notes:* For chewing-gum trials the first run began with an unchewed piece and the same gum was kept for all subsequent motions. For biscuit trials each run started with an empty, closed mouth; the subject then placed a biscuit, chewed as instructed, and swallowed.

Data processing. To reduce the noise, we apply a 4th-order butterworth filter to the data. The cutoff frequency is set to 6Hz, as human mastication frequency is around 1Hz to 2Hz . The data is then transformed to the head reference frame using rotation matrices.

3 Results

3.1 Mimicking human jaw motion

- human jaw motion from motion capture
- results of PCA on human jaw motion
- show graphs of human jaw motion vs robotic jaw motion
- show graphs of the force during chewing for human vs robot

3.2 chewing force

- max force that can be applied by the robot (both vertical and shear)
- show the force applied by the robot during chewing ?
- show the force applied by the human during chewing
- show the difference between the two

4 Discussion

4.1 Summary of findings

4.2 limitations

- So far very big and heavy robot due to steel plates and big actuator != human jaw
- 3D printed teeth/jaw not strong enough to withstand the forces applied by the actuators

4.3 Future work

- 3D printed jaw/teeth to be replaced by a more rigid material
- add a tongue module
- add a saliva module
- adapt state machine to coordinate the different modules
- add a camera to track the food

5 Conclusion

6 References

[1] Steven Mills et al. "Principal Component Representations of Chewing Motion". In: IVCNZ '14 (2014), pp. 218–223. DOI: 10.1145/2683405.2683434. URL: https://doi.org/10.1145/2683405.2683434.

[2] Meg Simione et al. "Differing structural properties of foods affect the development of mandibular control and muscle coordination in infants and young children". In: *Physiology & Behavior* 186 (2018), pp. 62–72. ISSN: 0031-9384. DOI: https://doi.org/10.1016/j.physbeh.2018.01.009. URL: https://www.sciencedirect.com/science/article/pii/S0031938418300155.