ECE 445 / ME 470

SENIOR DESIGN LABORATORY

FINAL REPORT

Softbot: Jellyfish-Inspired Bionic Soft Robot with Visual Perception

---

Team #14

YINLIANG GAN
(gan10@illinois.edu)
JUNWEI ZHANG
(junweiz4@illinois.edu)
WANGJIE XU
(wangjie3@illinois.edu)
SHURAN YAN
(shurany2@illinois.edu)

TA: Jiangshan Zhuo

May 27, 2025

Abstract

The ocean, covering over 70% of the Earth's surface, is vital for climate regulation, resource provision, and human development. This importance has spurred the need for exploration and utilization of underwater environments, which present significant challenges due to their hazardous and dynamic nature. Traditional rigid actuators, despite their high power output, lack flexibility, adaptability, and are prone to mechanical failure in these unpredictable conditions. To address these limitations, soft robotics, offering increased flexibility and environmental compliance, has emerged as a promising solution. However, existing soft actuators, such as dielectric elastomer actuators (DEAs) and hydraulically amplified self-healing electrostatic (HASEL) actuators, still face challenges in deformation and force output, limiting their effectiveness in underwater robotics. This study presents a hybrid actuation system that combines the strengths of DEAs and HASEL actuators to enhance the performance of underwater robots. The integration of reinforcement learning (RL) and visual perception improves the robot's control and decision-making capabilities, enabling optimal performance in complex underwater environments. The system features three high-level functionalities: autonomous swimming with efficient propulsion, real-time environmental perception, and intelligent path planning and decision-making via RL. These capabilities are enabled by a novel hybrid actuation approach, a visual perception system for tracking the robot's position and environment, and an RL control system that adjusts movements based on feedback. The integration of these subsystems ensures robust, adaptable, and autonomous navigation in dynamic underwater environments.

Contents

1 Introduction

1.1 Purpose

The ocean, covering more than 70% of the Earth's surface, plays a critical role in climate regulation, resource provision, and human development [1]. These factors have led to the increasing need for exploration and utilization of underwater environments, which presents significant challenges due to their hazardous and unstructured nature [2]. In these environments, traditional rigid actuators, though high in power output, are limited by their poor flexibility, adaptability, and risk of mechanical failure, especially in dynamic and unpredictable conditions [3], [4]. To address these limitations, soft robotics has emerged as a promising solution, offering increased flexibility, compliance, and environmental adaptability [5]. However, current soft actuators, such as dielectric elastomer actuators (DEAs) and hydraulically amplified self-healing electrostatic (HASEL) actuators, still face challenges, such as limited deformation and force output, which restrict their effectiveness in underwater robotics [6]–[11].

The goal of this study is to combine the strengths of DEAs and HASEL actuators to develop a hybrid actuation system that leverages both technologies to enhance the performance of underwater robots. This hybrid approach will enable greater locomotion, environmental adaptability, and durability, addressing the challenges posed by traditional rigid actuators. Additionally, the integration of reinforcement learning (RL) and visual perception will improve the robot's control and decision-making capabilities, ensuring optimal performance in complex underwater environments [12]–[15].

1.2 Functionality

The system is designed to fulfill three high-level functionalities, each of which supports the overall goal of developing a soft, autonomous underwater robot with intelligent navigation and adaptive control.

1. Autonomous swimming with effective propulsion. The bionic jellyfish robot (2D-BJR) is capable of generating forward motion through the coordinated actuation of two electrohydraulic bent-to-straight actuators (EBSAs). These actuators enable rapid contraction and extension, mimicking natural jellyfish pulsing. Achieving a reasonably high swimming speed demonstrates that the hybrid actuator system delivers both power and compliance, aligning with the project's objective to develop efficient and biomimetic underwater locomotion.

2. Visual perception and environmental responsiveness. An onboard visual processing system allows the robot to perceive its surroundings in real-time. Through an integrated neural network, it extracts key information such as position, orientation, and external changes in the environment. This enables the robot to make context-aware decisions, supporting the broader goal of deploying autonomous systems in unstructured underwater environments.

3. Path planning and decision-making via reinforcement learning. The robot uses a

reinforcement learning (RL) controller to plan a sequence of waypoints and navigate toward the target. The RL agent receives continuous feedback from the visual system and adjusts motor outputs accordingly. This capability reflects the project's goal of achieving closed-loop intelligent control that allows for robust, adaptable, and autonomous navigation in dynamic settings.

1.3 Subsystem Overview

The system consists of several key subsystems, each playing a vital role in realizing the intended functionality of the 2D-BJR.

Actuation System: This system adopts a novel hybrid actuation approach that combines the advantages of DEAs and HASEL actuators. It enables precise control over deformation, while also offering large strain capability and self-healing properties. This combination enhances the flexibility and durability of the underwater robot.

Visual Perception System: The visual network processes images captured by the camera to track the robot's position, velocity, and orientation. This information is essential for real-time motion adjustment and accurate tracking of the predefined trajectory.

Reinforcement Learning Control System: The reinforcement learning (RL) control module continuously optimizes motion decisions based on visual feedback, enabling the robot to adapt more effectively and perform robustly in dynamic environments.

Control System: The control system receives motion commands generated by the RL algorithm and transmits them to high-voltage relays to execute corresponding movements. This ensures that the 2D-BJR can precisely adjust its motion to follow the predefined path.

Figure 1 illustrates the interactions among these subsystems, highlighting the integration of actuation, visual perception, and reinforcement learning control in the 2D-BJR.

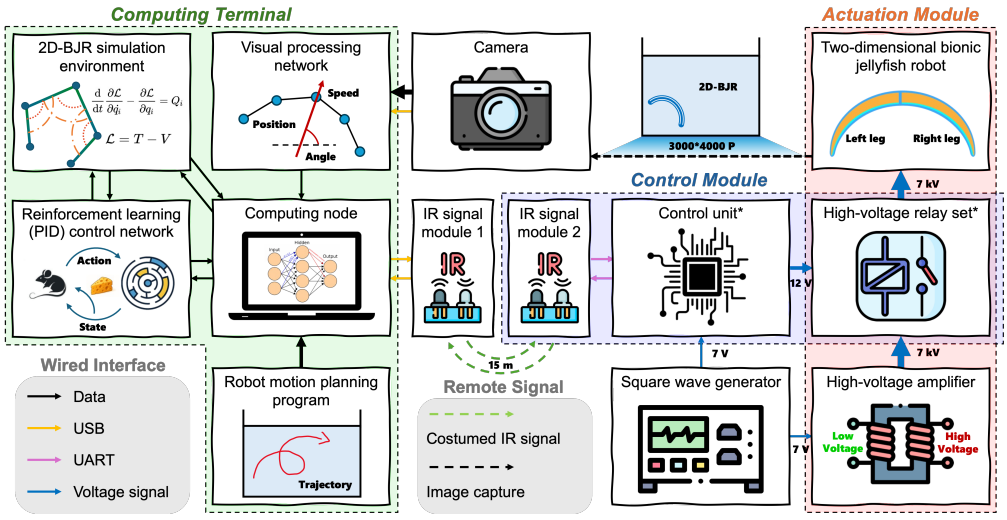


Figure 1: Block Diagram of 2D-BJR system

2 Design

2.1 Design Procedure

2.1.1 Design Decisions & Alternatives

Simulation environment [Classical Newton / Lagrange Analysis]: We tried both in modeling process and chose the Lagrange formulation for the simulation environment because it provides a more convenient approach for modeling complex systems involving constraints, which is just what 5-point model is .

RL control network [DQN / PPO]: We selected Proximal Policy Optimization (PPO) over Deep Q-Networks (DQN) because PPO is known for its stability and efficiency in continuous action spaces, which is crucial for fine-tuned control of robotic movements in dynamic underwater environments. And the training results further proved this.

Computing node [In terminal / GUI]: We opted for the Graphical User Interface (GUI) version because it provides a more user-friendly interface for real-time monitoring and control, which is essential for debugging and visualizing the robot's behavior.

Path planning program [BFS / A*]: We decided on the A* algorithm for path planning as it offers efficient navigation, especially in complex, dynamic environments like underwater exploration, where BFS proved to be too slow and resource-intensive.

IR module [Off-the-shelf module / Custom module]: We chose custom module and design by ourselves, since the off-the-shelf module cannot meet our encoding and decoding requirement, due to its weak noise filtering capabilities.

Object Detection Backbone [YOLOv8 / YOLOv5]: YOLOv8 is used for visual perception due to its high accuracy, fast inference speed, and lightweight design, making it suitable for real-time underwater tracking. Compared to YOLOv5, it introduces C2f and SPPF modules to enhance feature reuse and multi-scale aggregation, improving detection robustness in complex environments. The model includes a backbone for feature extraction, refinement modules, and a detection head for bounding box and class prediction.

Control Unit [Arduino chip updates / Four types of control outputs / Two types]: We update Arduino chip to a faster chip [ESP32 version], since the Ardunio Every version is to slow to handle the monitor refreshing.

Actuation Method [HASEL / DEAs / Combination]: In this work, we adopt a hybrid actuation approach that combines the advantages of both DEAs and HASEL actuators. This integration serves as one of the main innovations of our study.

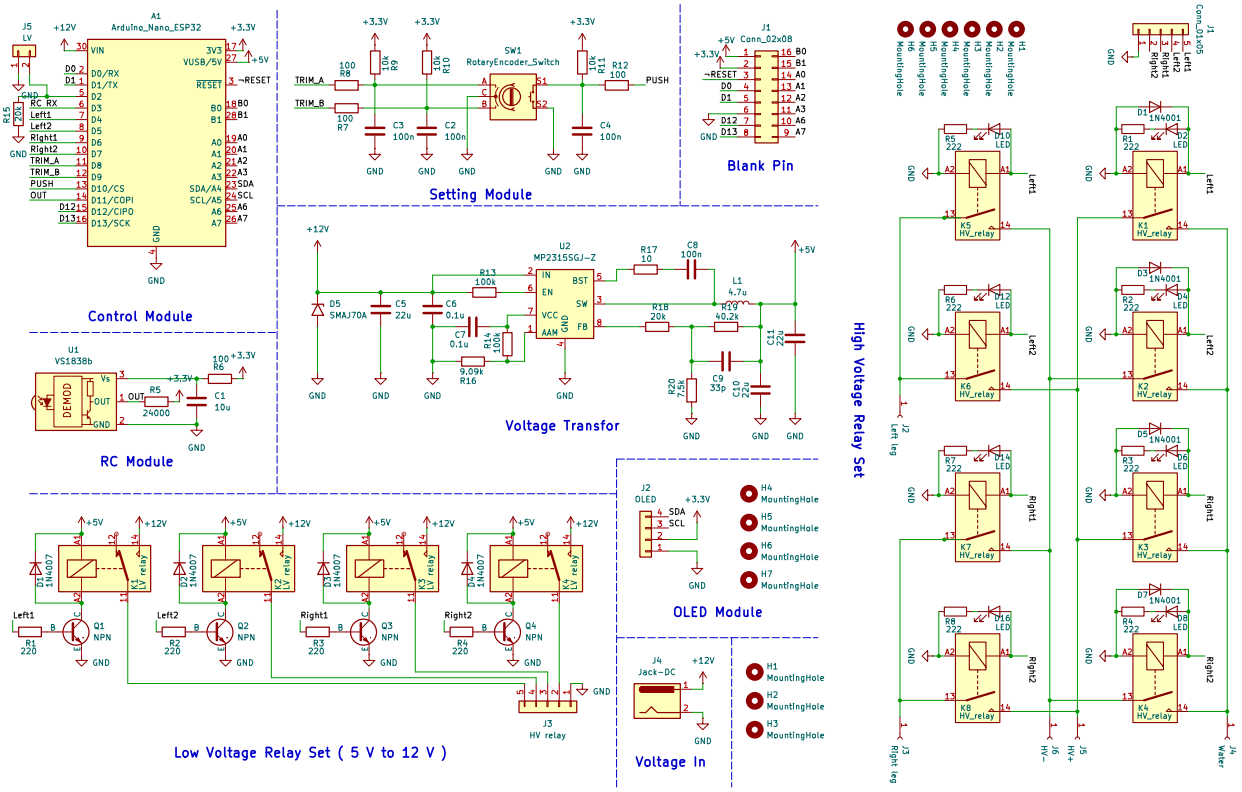
High-Voltage Relay Set [Passive Discharge / Polarity Switching / Active Discharge]: We choose the active discharge method, as the passive discharge approach is too slow to meet the actuation speed requirements of soft robots, and the polarity switching method, which uses water as the positive electrode, poses safety risks and is the most difficult to implement reliably.

2.1.2 Design Tools

We used Python as the main programming language, employing PyQt5 for GUI development, Matplotlib/NumPy/SymPy for computation and modeling, Stable-Baselines3 for reinforcement learning, Arduino IDE for microcontroller programming, KiCad for PCB design, and Fusion 360 for mechanical modeling.

2.1.3 General Form of the Circuits

There are generally two parts of circuit design in our project, control unit's circuit and HV relay set's circuit.



(a) Control unit (b) HV relay set

Figure 2: Schematic diagram of the circuits

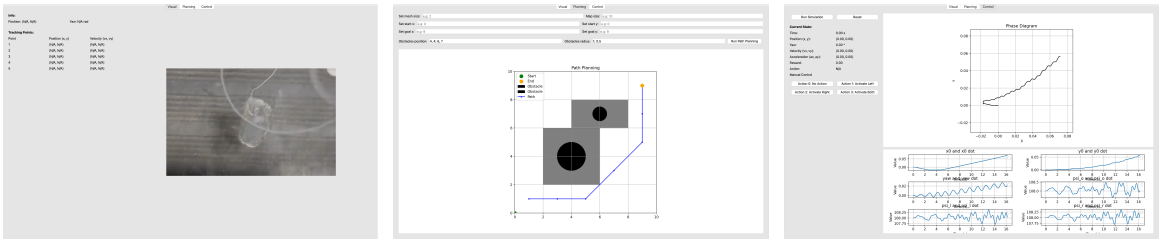
2.2 Design details

2.2.1 Computing Terminal

Computing Node

The computing node is a software system designed to handle the entire workflow of jellyfish control, from processing real-time video input to generating control signals. It uses a camera to capture video, which is then analyzed by the YOLO object detection model for identifying relevant objects. Based on this data, the path planning module calculates the

optimal movement, while the reinforcement learning module, trained by the iterative interaction with the jellyfish simulation environment, refines decision-making to improve control strategies. The system visualizes the data, display the workflow of the control algorithm with GUI, and the final control signals are sent to the control unit via an RC-sender using the UART and NEC protocol. As illustrated in Figure 3, we developed a control system with a graphical user interface (GUI) using the PyQt5 library, enabling intuitive and real-time interaction with the system. It allows real-time monitoring of visual recognition results, execution of path planning program, RL / manual control in simulation environment and IR signal sending process.



(a) Visual perception results (b) Path planning setting (c) Control gui setting

Figure 3: GUI for Control System

Visual Processing Network

To enable robust tracking of soft-bodied underwater robots, we implement a modular multi-object tracking (MOT) pipeline. The system integrates both motion and appearance cues to enhance tracking accuracy in dynamic underwater environments.

The pipeline begins with object detection using a pretrained YOLOv8 model. Detected bounding boxes are processed by two parallel modules: a motion-based prediction path using OC-SORT (with Kalman filtering and camera motion compensation), and an appearance-based association path using a Re-Identification (ReID) network. Domain adaptation techniques are optionally applied to improve robustness under underwater image variability.

These two cues are fused via an affinity weighting mechanism to construct a cost matrix, which is then solved using the Hungarian algorithm for optimal object-to-track assignment. This design allows consistent identity tracking over time, even under occlusion or deformation, making it well-suited for soft, deformable agents.

Path Planning Program

The Motion Planning Program is designed to generate optimal paths for robotic movement in environments with obstacles, specifically for the jellyfish robot. Users can input the robot's starting point, target endpoint, and the positions of circular obstacles in the environment. Using this information, the program calculates a trajectory that allows the jellyfish robot to safely reach its destination while avoiding obstacles.

The program discretizes the environment into a grid-based map with adjustable grid sizes, making it suitable for A* pathfinding. The grid allows the space to be represented as

a series of cells, where each cell can either be traversable or obstructed by an obstacle. The A* algorithm is then applied to compute the shortest path, using the key formula,

$$f(n) = g(n) + h(n), h(n) = \sqrt{(x_n - x_{\text{goal}})^2 + (y_n - y_{\text{goal}})^2} \tag{1}$$

where $f(n)$ is the total estimated cost of the cheapest solution through node n, $g(n)$ represents the actual cost from the start node to node n, and $h(n)$ is the heuristic estimate of the cost from node n to the goal. We design the heuristic cost $h(n)$ as the Euclidean distance between the current node n and the goal node:

Jellyfish Simulation Environment

Lagrangian mechanics is a reformulation of classical mechanics that provides a powerful method for analyzing the motion of systems. It is based on the principle of least action, which states that the path taken by a system between two states is the one that minimizes the action, a quantity defined as the integral of the Lagrangian over time. The key principle is as the formula below:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i}\right) - \frac{\partial \mathcal{L}}{\partial q_i} = Q_i \tag{2}$$

where $Q_i = \vec{F} \cdot \frac{\partial \vec{r}}{\partial q_i}$, $\mathcal{L} = T - V$, T is the expression for kinetic energy, V denotes the potential energy, $\vec{F}$ is the external force, $\vec{r}$ is the action point vector, q_i is the degree of freedom variables.

As shown in Figure 4(a), our jellyfish body, consisted of 5 mass points, 4 rods and 3 torque springs connecting the center of two adjacent poles, has six degrees of freedom.

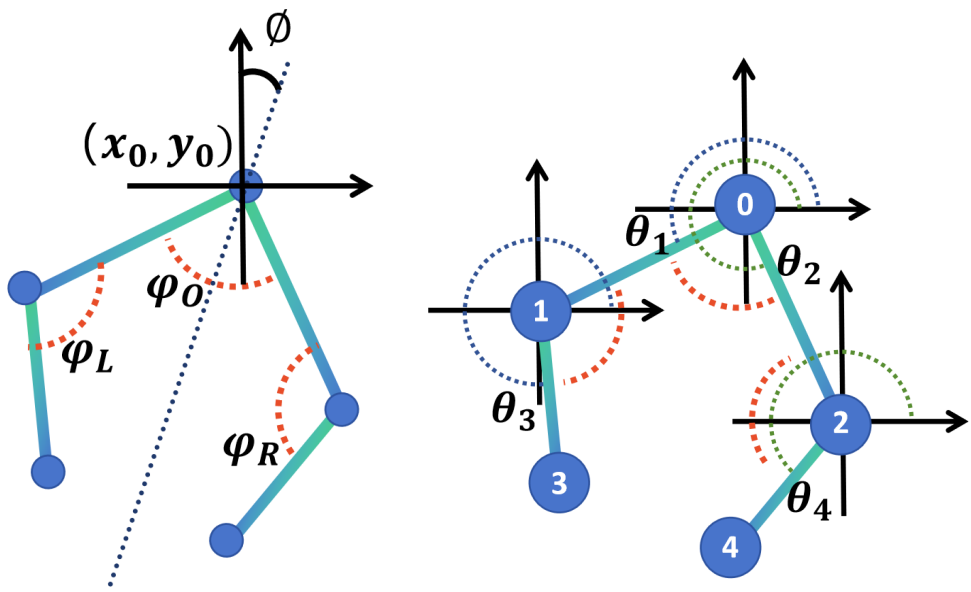
To derive the expression for kinetic energy, we first need to determine the positions of the five particles using the six degrees of freedom. The angles θ can also be expressed in terms of four angular degrees of freedom (for completeness, we define $\theta_0 = 0$):

$$\boldsymbol{\theta} = \begin{pmatrix} 0 \\ -\frac{1}{2} \\ \frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \end{pmatrix} \psi_o + \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} \psi_l + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \psi_r + \begin{pmatrix} 0 \\ -1 \\ -1 \\ -1 \\ -1 \end{pmatrix} \phi + \begin{pmatrix} 0 \\ \frac{3\pi}{2} \\ \frac{3\pi}{2} \\ \frac{\pi}{2} \\ \frac{\pi}{2} \end{pmatrix} \tag{3}$$

So the positions of the nodes can be expressed as:

$$x_1 = x_0 + Rcos(\theta_1), x_2 = x_0 + Rcos(\theta_2), x_3 = x_1 + Rcos(\theta_3), x_4 = x_2 + Rcos(\theta_4) \tag{4}$$

$$y_1 = y_0 + Rsin(\theta_1), y_2 = y_0 + Rsin(\theta_2), y_3 = y_1 + Rsin(\theta_3), y_4 = y_2 + Rsin(\theta_4) \tag{5}$$



(a) Annotation of 6 degrees of freedom

(b) Annotation of point number and angle

Figure 4: Schematic diagram of 5-point jellyfish robot modeling

After deriving the positions of the five nodes, we can compute their time derivatives to obtain the expression for the kinetic energy, as $T = \frac{1}{2}m\left(\dot{x}^T\dot{x} + \dot{y}^T\dot{y}\right)$.

For the potential energy, since we assume that the jellyfish robot will adjust to a suspended state and move in a two-dimensional horizontal plane, we only consider the elastic potential energy of the three springs in the dynamic process. With the potential energy formula for torque spring $V_{torque\_spring} = 0.5K(\theta - \theta_0)^2$,

$$V = 0.5K\left(\left(\psi_{l0} - \psi_l(t)\right)^2 + \left(\psi_{o0} - \psi_o(t)\right)^2 + \left(\psi_{r0} - \psi_r(t)\right)^2\right) \tag{6}$$

For simplification, we idealize the points of application of both external forces as the midpoints of each rod, with the direction perpendicular to the rod.

For the driving force F_D, in order to simulate the characteristic of "flattening" the jellyfish robot, we define the driving force directions for the left rods F_{D1} and F_{D3} to be clockwise, while the right rods experience counterclockwise forces. The magnitude of the activating force is constant and given by F_{D0}. Thus, the magnitude of the driven force is $F_{Di} = f_i F_{D0}$, where f_i is the off / on signal, be 1 when activated, otherwise be 0.

For the water reaction force F_W, in order to simulate the impact of the surrounding water during the jellyfish robot's contraction phase and to account for the typical form of resistance, we model $F_W = -Cv_\perp^2\hat{\mathbf{v}_\mathbf{w}}$, where $\hat{\mathbf{v}_\mathbf{w}}$ is the unit direction vector representing the direction of the velocity of the midpoint between the two nodes, along the direction of the component perpendicular to the rod, $v_\perp$ is the average speed of both ends of the pole.

After obtaining the differential equations for the six degrees of freedom using Lagrangian mechanics, we solve them numerically using the RK45 method.

2.2.2 Reinforcement Learning Control Network

To control the jellyfish robot's motion, we implement a deep reinforcement learning (DRL) approach, which combines reinforcement learning (RL) and deep learning. In RL, an agent interacts with an environment and learns to maximize cumulative rewards [16]. The Bellman equation governs this process:

$$Q^*(s,a) = E[r + \gamma \max_{a'} Q^*(s',a')] \tag{7}$$

Where $Q^*(s,a)$ is the optimal action-value function, r is the immediate reward, γ is the discount factor, s' and a' are the next state and action.

Given that the control output is binary (two-bit encoding), we employ Proximal Policy Optimization (PPO) [15], a policy gradient method effective in discrete action spaces. The PPO objective is:

$$L^{CLIP}(\theta) = \hat{E}_t \left[\min \left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t \right) \right] \tag{8}$$

Where $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$ is the probability ratio of the new and old policies, $\hat{A}_t$ is the advantage estimate at time t, ϵ is the clipping parameter.

To realize the control objective, the agent keeps updating and optimizing its policy to track these points pre-calculated by the path planning program sequentially until reaching the goal, as shown in Figure 5.

The state space T includes both visual recognition data (positions and orientations of five points) and the planned trajectory. The path planning system provides the XY coordinates of the trajectory points, D.

$$T = \begin{pmatrix} x_0 & y_0 & \phi_0 & \cdots \\ x_1 & y_1 & \phi_1 & \cdots \\ x_2 & y_2 & \phi_2 & \cdots \\ x_3 & y_3 & \phi_3 & \cdots \\ x_4 & y_4 & \phi_4 & \cdots \end{pmatrix}, D = \begin{pmatrix} tx_0 & ty_0 \\ tx_1 & ty_1 \\ \vdots & \vdots \\ tx_{n-1} & ty_{n-1} \end{pmatrix} \tag{9}$$

Where $x_i, y_i, \phi_i, i \in \{0,1,2,3,4\}$, represent the coordinates and orientations of the detected points, and $\cdot$ indicates motion features like velocity and angular velocity. tx_i, ty_i, $i \in \{0,1,\cdots,n-1\}$, represent the i-th target points, and n is the length of the path.

The agent's action space corresponds to the control of the jellyfish's electrodes using a binary encoding representing the on/off status of the left and right electrodes.

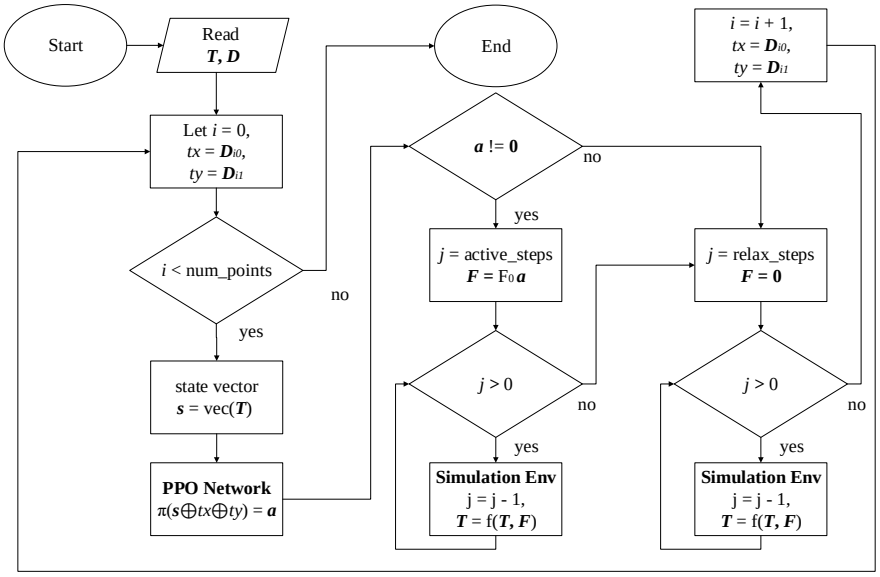


Figure 5: RL control procedure flowchart.

2.2.3 Control Module

Control Unit

The control unit decodes two-bit infrared signals and translates reinforcement learning (RL) outputs into control commands for a four-channel relay circuit, enabling directional movement of the jellyfish robot by activating the left and right electrodes.

To ensure reliable and consistent behavior, we implemented the control logic using a Moore state machine, where outputs depend solely on the current internal state. The machine accepts three binary inputs: **L** (left electrode activation), **R** (right electrode activation), and **C** (hold signal, 1 = continue, 0 = stop). Based on these, it generates two outputs: $\mathbf{O_L}$ and $\mathbf{O_R}$, which determine whether the left and right electrodes should be activated, respectively.

This updated logic improves stability and responsiveness during actuation. The full state transition process is visualized in Figure 6. The corresponding circuit schematic and PCB layout are shown in Figures 2(a) and 7, respectively.

HV Relay Set

To ensure that the soft robot can operate at a high frequency, we adopted a charge–discharge control strategy (see Fig. 8). In this configuration, Relay A is driven by a pulsed control signal, while Relay B is always driven with the inverse of A's signal. In **Phase 1**, Relay A is closed and Relay B is open. The high-voltage amplifier charges the electrohydraulic bent-to-straight actuators (EBSA) through a low-resistance path. According to the RC

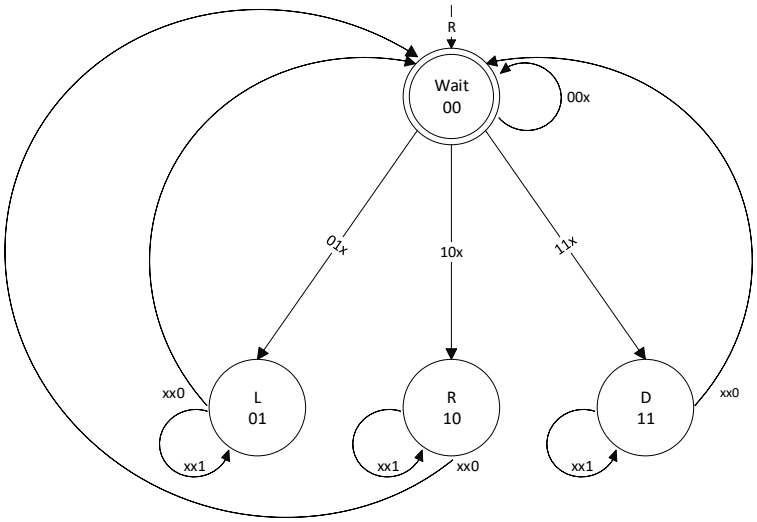
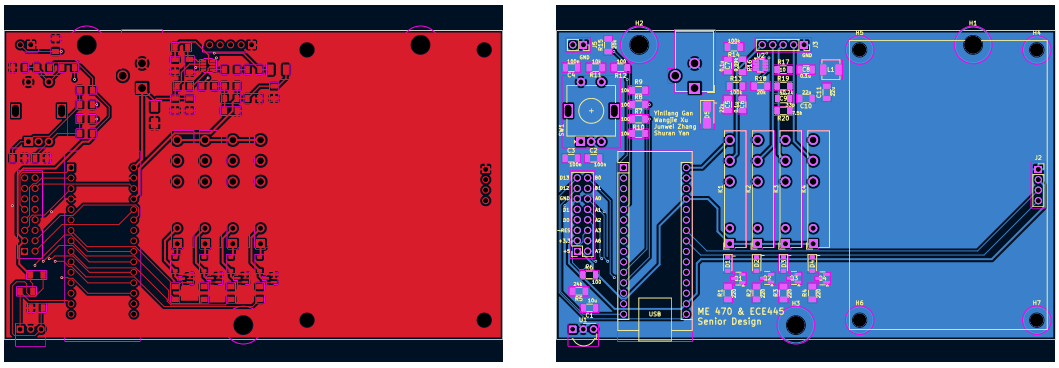


Figure 6: Updated Moore state machine diagram for electrode control.



(a) Front copper layer only (F.Cu). (b) All other PCB layers except F.Cu.

Figure 7: PCB layer visualization of the control unit. (a) Displays only the front copper layer (F.Cu), used for routing signal and power traces. (b) Shows all remaining PCB layers, including silkscreen, outline, vias, solder mask, and mechanical features.

charging equation:

$$t_{\text{charge}} = -\ln\left(\frac{V_{\text{target}}}{V_{\text{source}}}\right) \cdot R_{\text{sea}} C \tag{10}$$

Given a capacitance of $C = 0.2603\ \text{nF}$ and a charging resistance of $R_{\text{sea}} < 29.7\ \Omega$, the time to reach 99% of full charge is less than 35.4 ns. In **Phase 2**, Relay A opens and Relay B closes. The EBSA is then connected into an RC discharge loop that includes the seawater resistance and a discharge resistor. The total resistance is less than 1029.6 Ω. According to the RC voltage decay equation:

$$t_{\text{discharge}} = -\ln\left(\frac{V_0 - V_{\text{target}}}{V_0}\right) \cdot R_{\text{total}} C \tag{11}$$

The time required for the voltage to drop to 1% of its initial value is less than 1.23 μs. The charging and discharging times are sufficiently short to be negligible within the actuation

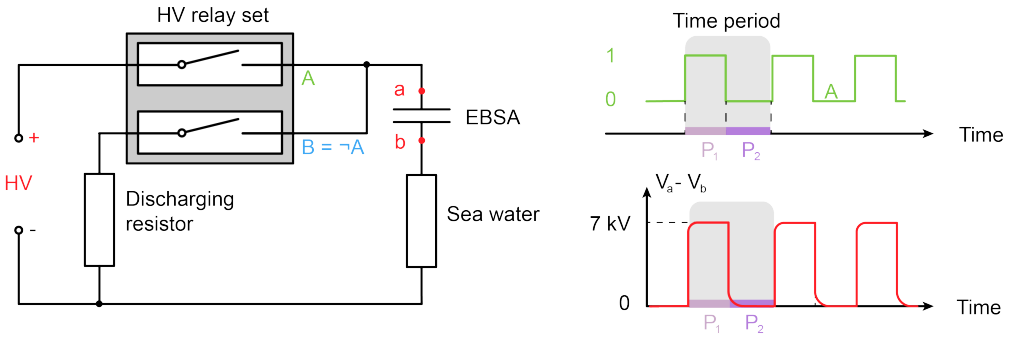


Figure 8: Two-relay controlled discharge strategy for safe and reversible EBSA actuation. A control cycle consists of two phases: a charging phase and a discharging phase. The discharge resistor used in this configuration is 1000 Ω.

cycle. Therefore, the redesigned two-relay circuit ensures complete charge and discharge of the EBSA in each cycle, enabling consistent and high-performance actuation.

2.2.4 Actuation Module

HV Amplifier

The HV Amplifier is used to amplify low-voltage signals to the HV required by certain system components, in our project, the actuation module. For this project, we will be using the MR HV amplifier provided by laboratory. This amplifier can output a voltage of up to ±10 kV with a maximum power of 20 W, suitable for HV power supply and testing.

HV Relay Set

As detailed in Section 2.2.3 ***HV Relay Set***, the implementation here follows the exact same design and logic, and is thus omitted for redundancy.

2D-BJR: Hydrogel Electrode Layer

This hydrogel is formed through photo-initiated free radical polymerization (triggered by I2959, crosslinking AAM with MBAA) to create a covalent network, while GDL-mediated pH adjustment promotes calcium ion release from hydroxyapatite (HA), inducing ionic crosslinking with alginate (ALG). The resulting dual covalent-physical network combines high mechanical strength with dynamic tunability.

According to the theorem of crosslinking, we guess that the proportion of MBAA, AAM, and ALG have great impact on the property of the hydrogel. As Table 1, for the first two hydrogel samples, we change the proportion of MBAA and let other materials remain the same. We find that as the proportion of MBAA increases, the modulus of the hydrogel increases a lot, while the max strain of the sample is reduced from 4.6 to 1. The yield stress increases a little bit. For the second and third samples, we change the proportion of ALG, and let other materials the same as sample 1. We find that if the quantity of ALG is increased the stickiness of the solution will increase, which leads to too many bubbles

Table 1: Results of the stretching experiment for the hydrogel electrode layer.

	Sample 1	**Sample 2**	**Sample 3**	**Sample 4**	**Sample 5**	**Sample 6**
Water	25 mL	25 mL	25 mL	25 mL	25 mL	25 mL
MBAA	0.0475 g	0.0085 g	0.0475 g	0.0475 g	0.0475 g	0.0475 g
AAM	4.825 g	4.825 g	4.825 g	4.825 g	3.5 g	6.0 g
I 2959	0.035 g	0.035 g	0.035 g	0.035 g	0.035 g	0.035 g
ALG	0.6025 g	0.6025 g	1.205 g	0.30125 g	0.30125	0.30125 g
HA	0.02 g	0.02 g	0.02 g	0.02 g	0.02 g	0.02 g
GDL	0.115 g	0.115 g	0.115 g	0.115 g	0.115 g	0.115 g
Yield stress	33 kpa	30 kpa	29 kpa	77 kpa	26 kpa	59 kpa
Modulus	51 kpa	12 kpa	65 kpa	57 kpa	26 kpa	59 kpa
Max strain	105%	462%	69%	203%	144%	196%

and make trouble in solidification. If we reduce the quantity of ALG, there is no bubbles, and the stretch properties of the sample is excellent. The yield stress and modulus are large enough, while the max strain is also acceptable. Lastly, we change the proportion of AAM through making hydrogel 5 and 6. From the result we can see that the reduction of AAM has negative effect on the stretch property, since all the properties is reduced. The increasing of AAM does not have a large impact on the stretch property, so we think the increase of AAM is unnecessary.

2D-BJR: Main Body

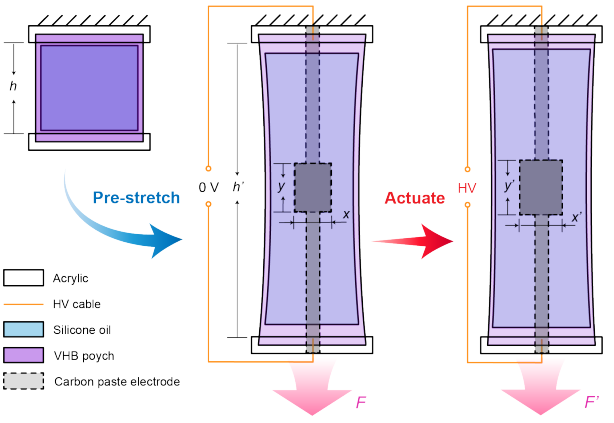


Figure 9: Illustration of the pre-stretching experiment for the DE-HASEL actuator.

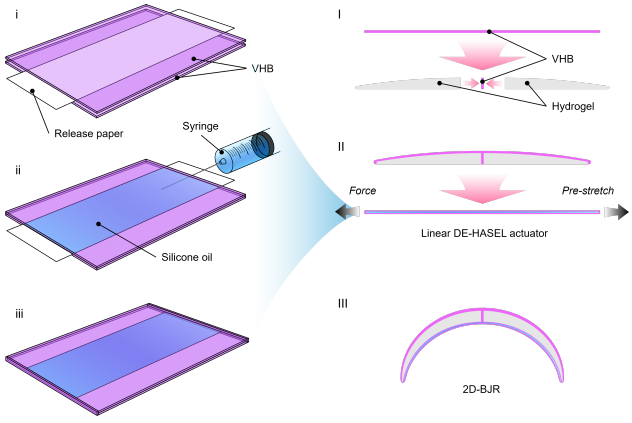


Figure 10: Manufacture process of 2D-BJR.

To determine the optimal pre-stretching parameters for fabricating EBSAs, we characterized the linear DE-HASEL actuator under various pre-stretch ratios. As shown in Figure 9, the experiment was conducted as follows: the original length h of the fabricated DE-HASEL actuator was first measured. Then, the actuator was pre-stretched using a custom-designed stretching apparatus, and its stretched length h' was recorded. The stretch ratio was calculated as h'/h.

Subsequently, the actuator's electrode dimensions were measured, including the length x along the stretching direction and the width y perpendicular to it, along with the applied force F during stretching. Afterward, the actuator was electrically activated, and the stretched dimensions x', y' and the actuation force F' were recorded. From these measurements, we derived the axial stretch ratio x'/x, the transverse stretch ratio y'/y, and the force change $F - F'$ under different pre-stretch ratios and driving voltages, as shown in Figure B.12.

Figure 10 shows the fabrication process of the linear DE-HASEL actuator, the core of the 2D-BJR. Two rectangular VHB films were prepared, with a smaller release paper placed between them. The exposed VHB edges were pressed and bonded, forming a sealed pouch. Silicone oil was injected into the pouch and evenly spread, after which the release paper was removed and air bubbles expelled. The opening was then sealed with another VHB layer.

To assemble the 2D-BJR, two shaped hydrogel substrates were used. Unstretched VHB was applied to one side of a hydrogel, and the pre-stretched actuator was bonded to the other side. Upon releasing the prestretch, the structure curled into a jellyfish-like shape, completing the 2D-BJR.

2.2.5 Peripherals

The peripheral modules of the system include several key components. The IR signal module enables wireless communication between the computing node and the control unit by transmitting infrared signals and converting them via a TTL-to-USB cable. The camera captures real-time video input, providing essential data for object detection and visual tracking. The square wave generator, specifically the AFG1022 model, produces stable signals with configurable frequency to drive the actuator effectively. Finally, the glass water tank serves as a transparent and sealed testing environment for the jellyfish robot, ensuring both experimental integrity and clear video monitoring.

3 Design Verification

3.1 Completeness and Verification of Requirements

The high-level requirements of the system comprehensively cover key functional modules, including the actuation system, visual perception, and intelligent control, with clearly defined performance targets for each. The actuator is required to provide strong and stable force feedback; the visual system must process environmental inputs in real time with

high precision; and the reinforcement learning controller should be capable of guiding the robot along a predefined path. Each requirement is verified through quantitative testing: actuator performance is evaluated through motion analysis, the visual system is tested based on object recognition accuracy and response delay, and the reinforcement learning module is assessed via path-tracking accuracy and task success rate. Detailed low-level requirements are provided in Appendix A.

3.2 Quantitative Results

Quantitative results for all requirements are provided below. These results are consistent with the verification procedures, ensuring that the system meets or exceeds the defined specifications.

Measured Technical Parameters: The following table lists the technical parameters directly listed in the R&V table in Appendix A. The results in Table 2 are the average results obtained through more than 10 measurements.

Table 2: Verification results of key system requirements

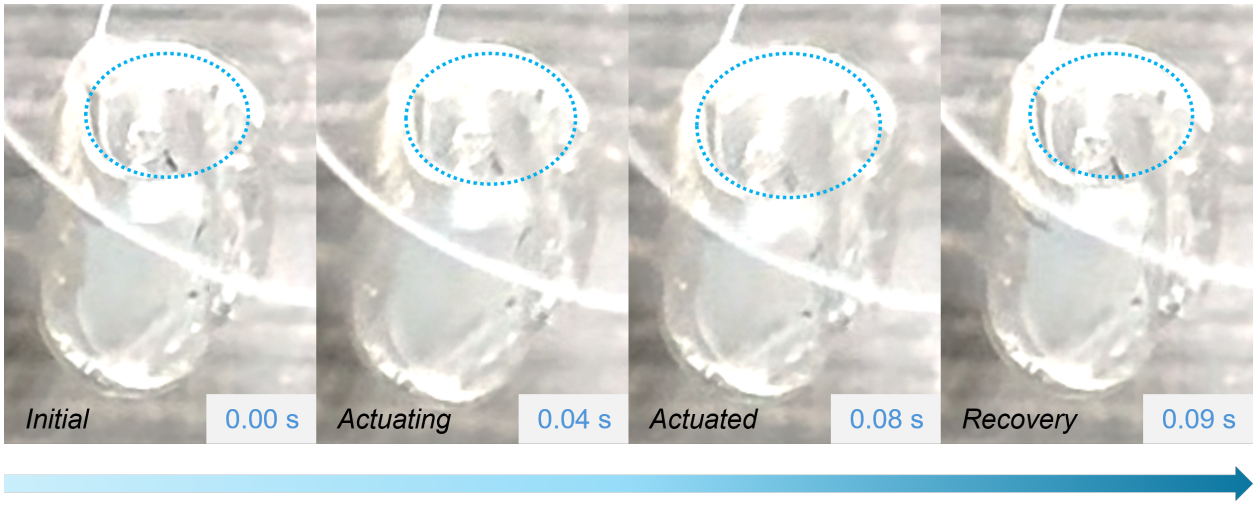
Requirement	**Target**	**Result**
Data process rate	$\geq$ 30fps	41fps
Computer vision accuracy	$\geq$ 95%	98.3%
Computing node delay	$\leq$ 200ms	73ms
Target reaching precision	$\leq$ 2radii	1.72radii
RL control speed	$\leq$ 30fps	39fps
Path planning speed	$\leq$ 30fps	202fps
Control unit processing time	$\leq$ 150ms	57ms
IR signal delay	$\leq$ 100ms	22ms

RL Training Results: Figure B.15 presents the training curve of the RL agent. Gradually rising and converging curve indicates that it has found a better strategy. Figure B.14 shows the control process of the RL agent - the agent first activates the left, adjusting the direction, then activates both, controls the jellyfish bot move straightly to the goal.

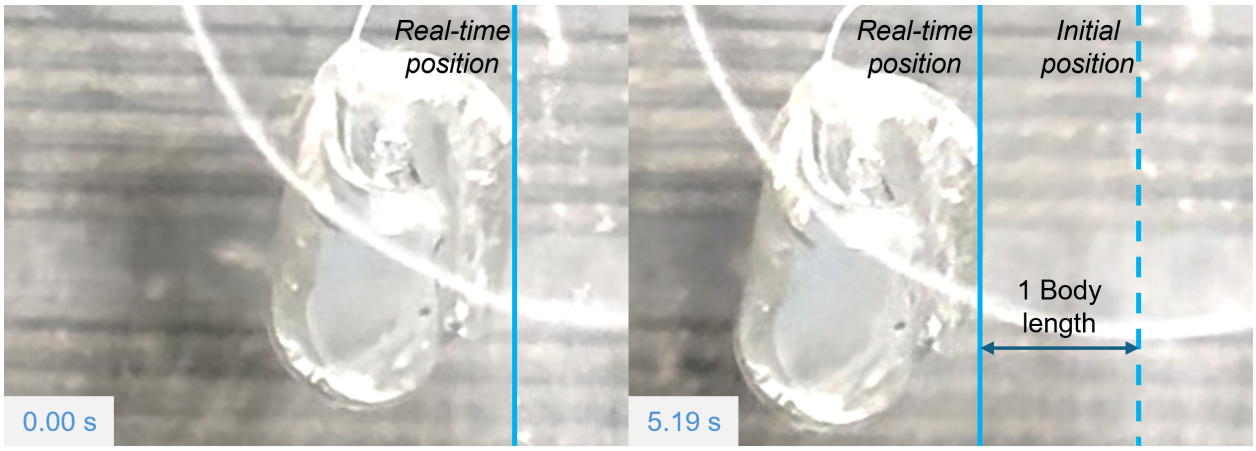
Linear DE-HASEL Actuators Results: As shown in Figure B.12, the results indicate that a higher pre-stretch ratio leads to improved performance of the linear DE-HASEL actuator. However, since VHB material tends to delaminate or rupture when stretched beyond 700%, we finally selected a pre-stretch ratio of 600% as the optimal value.

2D-BJR Swimming Results: Figure 11(a) illustrates a single actuation cycle of the 2D-BJR during forward motion. In each cycle, the actuator extends over approximately 0.08 s and rapidly contracts in about 0.01 s (actuation period = 0.15 s, i.e., 4 Hz). The contraction is significantly faster than the extension, enabling net forward propulsion.

Figure 11(b) demonstrates the locomotion of the 2D-BJR at 4 Hz and 9 kV. After 5.19 s of actuation, the robot travels nearly one body length, achieving a speed of approximately 0.188 BL/s —- substantially faster than the typical swimming speed of Aurelia aurita (0.08 BL/s).



(a) Single actuation cycle of the 2D-BJR, illustrating the extension and rapid contraction phases.



(b) Sequential locomotion of the 2D-BJR over nearly one body length under 4 Hz actuation at 9 kV.

Figure 11: Demonstration of the 2D-BJR's actuation and locomotion performance.

Hydrogel Electrode Layer Results: As showen in Figure B.13, after changing various proportion of the materials, we find that hydrogel 4 has the best stretch properties. It has very large modulus and yield stress. And it also has a reasonable max strain.

4 Costs & Schedule

4.1 Parts

The cost estimate in Table 3 covers both core robotic components and necessary testing infrastructure, including sensors, UV curing systems, and CNC machining. As a research-oriented prototype, many expenses reflect one-time development and validation efforts. In future iterations, costs could be significantly reduced through bulk purchasing, design simplification, and hardware reuse—potentially lowering total expenses by 40–60%.

Table 3: Representative Parts and Costs Summary

Item	**Description**	**Quantity**	**Price (yuan)**
Silicone oil	0.65 cSt, 1 L	1	39
VHB tapes	3M 4095, 1 mm and 0.5 mm	2	246
Hydrogel sheets	58×87 mm and 80×80 mm	88	190
High voltage relays	CRSTHV-20KV-A, PCB footprint	9	1653
Camera	12 MP, USB-A	1	227
Glass tank	200×100×10 mm quartz glass	1	180
Arduino	Nano Every, ATMega4809	1	120
Control PCBs	Custom-made, control + relay sets	15	566
CNC machining	Custom mechanical components	7	900
Sensors	Tension sensor + DAQ card	2	474
Misc. components	Resistors, sockets, cables, etc.	—	300
UV curing system	365 nm, 240 W	1	936.32
Total	—	—	**4831.32**

4.2 Labor

Assuming each team member worked 40 hours per week for 6 weeks, the total estimated labor cost is $\$120,000$, calculated using the formula Labor Cost $=$ Hourly Rate $\times$ Hours Spent $\times$ 2.5.

4.3 Schedule

Table 4: Project Timeline (April 14 – May 23, 2025)

Week	Date (2025)	Detailed Tasks	Member(s)
1	Apr 14–16	Prepare initial hydrogel samples	Junwei Zhang
	Apr 17–20	Preliminary mechanical and water absorption tests	Junwei Zhang
	Apr 14–18	Fabricate initial DE-HASEL actuators	Yinliang Gan
	Apr 19–20	Initial optimization of actuator fabrication	Yinliang Gan
	Apr 14–20	Capture ∼700 images of 3D-printed jellyfish	Shuran Yan
	Apr 14–20	Refine jellyfish simulation model and bug fixing	Wangjie Xu
2	Apr 21–24	Detailed mechanical tests (elastic modulus, fatigue life)	Junwei Zhang
	Apr 25–27	Electrical conductivity tests	Junwei Zhang
	Apr 21–27	DE-HASEL actuator prestretch experiments	Yinliang Gan
	Apr 21–27	Train preliminary detection and ReID algorithms	Shuran Yan
	Apr 21–27	Visualize data and conduct initial RL training	Wangjie Xu
3	Apr 28–30	Finalize optimal hydrogel formula and verify properties	Junwei Zhang
	May 1–4	Fabricate hydrogel electrodes	Junwei Zhang
	Apr 28–May 2	Assemble actuator with hydrogel electrodes	Yinliang Gan
	May 3–4	Initial jellyfish robot structural assembly	Yinliang Gan
	Apr 28–May 4	Refine dataset and algorithms; start angle calculation	Shuran Yan
	Apr 28–May 4	RL optimization and begin path planning scripts	Wangjie Xu

Continued on next page

Table 4 Continued from previous page

Week	Date (2025)	Detailed Tasks	Member(s)
4	May 5–8	Waterproof sealing	Yinliang Gan
	May 9–11	Finalize robot assembly	Yinliang Gan
	May 5–11	Complete angle calculation algorithm	Shuran Yan
	May 5–11	RL tracking for arbitrary targets and debug path planning	Wangjie Xu
5	May 12–15	Produce final datasets integrating real robot movements	Yinliang Gan, Shuran Yan
	May 16–18	Verify jellyfish robot line-following performance	Yinliang Gan, Shuran Yan
	May 12–18	Achieve sequential target RL tracking; start FSI simulations	Wangjie Xu
6	May 19–23	Integrate visual recognition, RL control, and physical actuation; validate infrared communication; conduct complete system testing	All Members

5 Conclusion

5.1 Accomplishments

Our project successfully fulfilled all high-level requirements outlined at the beginning of the design. First, the hybrid actuation system based on electrohydraulic bent-to-straight actuators (EBSAs) demonstrated both strong force output and structural stability. By combining dielectric elastomer actuators (DEAs) with HASEL actuators, we achieved enhanced compliance, controllability, and deformation, surpassing the performance of conventional soft actuators in underwater conditions.

Second, the onboard visual processing network—built upon the YOLOv8 object detection algorithm and a multi-object tracking (MOT) framework—enabled the system to robustly identify and track the jellyfish robot in real time. Accurate extraction of key position and orientation data allowed for adaptive feedback control, even in environments with dynamic lighting and water turbulence.

Third, our reinforcement learning (RL) control network, based on the Proximal Policy Optimization (PPO) algorithm, successfully generated discrete control commands to guide the robot toward predefined waypoints. The control system demonstrated reliable trajectory-following behavior in the test tank, with consistent loop closure between prediction and physical actuation.

Together, these accomplishments validate the effectiveness of our integrated design approach. The final 2D-BJR system combines innovative actuation, perception, and control into a cohesive platform capable of intelligent navigation in aquatic environments, which representing a significant step forward in the development of bioinspired underwater robots.

5.2 Uncertainties

Despite the successful realization of all high-level requirements, several uncertainties remain in the system's closed-loop performance due to the inherent limitations of soft robotic components and control latency. The most critical challenge lies in precise trajectory tracking under real-world disturbances, such as actuation variability and sensing noise.

Specifically, the actuation module introduces granularity and variability in displacement: each discrete pulse contributes approximately $\Delta x = 2.5\ \mathrm{mm}$, with up to $\pm 10\%$ deviation due to mechanical backlash and fluid interaction. Additionally, the control system experiences a delay of $\tau = 2 \pm 1$ timesteps, affecting real-time responsiveness. Sensor noise, with a bounded error of $\pm 2\ \mathrm{mm}$, further contributes to uncertainty in pose estimation.

To quantify the impact of these uncertainties, we constructed a mathematical model and conducted a parameter sweep simulation. The worst-case error propagation was derived from:

$$\Delta e_{t+1}^{\text{worst}} = \pm \delta_r - (n_{t-\tau} \pm 0.5)(\Delta x \pm 10\%) \tag{12}$$

where δ_r is the maximum rate of change in the reference trajectory. Simulation results demonstrated that even under worst-case conditions, the peak tracking error remained below $15\ \mathrm{mm}$, which is within the system's acceptable tolerance band of $\pm 20\ \mathrm{mm}$.

However, error increases with control delay and command quantization. These results highlight the need for future improvements in closed-loop feedback design, such as higher-resolution actuation or predictive compensation methods. Furthermore, while the system avoids sustained oscillation within the tested bounds, its robustness in more complex or larger-scale underwater environments remains to be evaluated.

5.3 Ethical Considerations

5.3.1 Adherence to IEEE Code of Ethics

This project fully adheres to the IEEE Code of Ethics by:

Public Safety and Responsible Research: All testing was conducted in controlled environments. The soft robot's compliant design eliminates mechanical injury risks and minimizes environmental disruption.

Data Privacy and Integrity: The vision system is used solely for navigation and control. No personally identifiable information (PII) is collected. Data is anonymized, and transmission is encrypted using TLS.

Scientific Honesty and Accuracy: All results are derived from reproducible experiments and validated simulations. Safety margins are computed and documented, avoiding exaggerated claims of performance.

5.3.2 Environmental and Data Ethics

Environmental Impact The robot is constructed using biodegradable hydrogels and elastomers certified for non-toxicity. No harmful microplastics or lubricants are released during operation. Environmental risk is modeled as $R = I \cdot P_f$, where potential impact I and failure probability P_f are minimized through careful design and lab validation. Low-Reynolds-number propulsion ensures minimal turbulence and avoids disturbing aquatic ecosystems.

Privacy Protection Cameras are used strictly for system feedback. No human subjects are recorded. All images are hashed and stored locally. No secondary usage or external distribution is permitted. Data minimization is enforced: only necessary frames are retained for navigation.

5.3.3 Safety Measures

Electrical Safety: Though the system uses high voltage (up to 7 kV), the stored energy is extremely low ($< 2\ \mu C$), and the charging time is limited to nanoseconds. This ensures no risk of electric shock or thermal hazard.

Mechanical Safety: The robot's maximum actuation force (≤ 5 N) and contact pressure (≤ 50 kPa) are well within human-safe thresholds.

Environmental Safety: All materials degrade within 2–3 weeks in marine environments (ASTM D6691). Sealants and coatings are water-stable and non-toxic.

Fail-Safe Protocols: Emergency kill-switches, watchdog timers, and flotation mechanisms are implemented. All team members completed lab safety training and follow a checklist before each test.

5.4 Future Work

Future improvements include reducing the control latency by integrating onboard computing modules, enhancing waterproof sealing for open-water deployment, and increasing energy efficiency through custom low-power HV amplifiers. Additional work will focus on expanding the robot's degrees of freedom, introducing 3D motion capability, and integrating self-sensing electrodes to enable closed-loop proprioception. Long-term, we envision this system being adapted for environmental monitoring or marine life interaction.

References

[1] S. Aracri, F. Giorgio-Serchi, G. Suaria, *et al.*, "Soft robots for ocean exploration and offshore operations: A perspective," *Soft Robotics*, vol. 8, no. 6, pp. 625–639, 2021, PMID: 33450174. DOI: 10.1089/soro.2020.0011. eprint: https://doi.org/10.1089/soro.2020.0011. [Online]. Available: https://doi.org/10.1089/soro.2020.0011.

[2] J. Yuh and M. West, "Underwater robotics," *Advanced Robotics*, vol. 15, no. 5, pp. 609–639, 2001. DOI: 10.1163/156855301317033595. eprint: https://doi.org/10.1163/156855301317033595. [Online]. Available: https://doi.org/10.1163/156855301317033595.

[3] J. Shintake, V. Cacucciolo, D. Floreano, and H. Shea, "Soft robotic grippers," *Adv. Mater.*, vol. 30, no. 29, p. 1 707 035, 2018. DOI: https://doi.org/10.1002/adma.201707035. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/adma.201707035. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.201707035.

[4] P. Polygerinos *et al.*, "Soft robotics: Review of fluid-driven intrinsically soft devices; manufacturing, sensing, control, and applications in human-robot interaction," *Adv. Eng. Mater.*, vol. 19, no. 12, p. 1 700 016, 2017. DOI: https://doi.org/10.1002/adem.201700016. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/adem.201700016. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/adem.201700016.

[5] W.-S. Chu, K.-T. Lee, S.-H. Song, *et al.*, "Review of biomimetic underwater robots using smart actuators," *International Journal of Precision Engineering and Manufacturing*, vol. 13, no. 7, pp. 1281–1292, Jul. 2012, ISSN: 2005-4602. DOI: 10.1007/s12541-012-0171-7. [Online]. Available: https://doi.org/10.1007/s12541-012-0171-7.

[6] A. O'Halloran, F. O'Malley, and P. McHugh, "A review on dielectric elastomer actuators, technology, applications, and challenges," *Journal of Applied Physics*, vol. 104, no. 7, p. 071 101, Oct. 2008, ISSN: 0021-8979. DOI: 10.1063/1.2981642. eprint: https://pubs.aip.org/aip/jap/article-pdf/doi/10.1063/1.2981642/15020155/071101_1_online.pdf. [Online]. Available: https://doi.org/10.1063/1.2981642.

[7] U. Gupta, L. Qin, Y. Wang, H. Godaba, and J. Zhu, "Soft robots based on dielectric elastomer actuators: A review," *Smart Materials and Structures*, vol. 28, no. 10, p. 103 002, Sep. 2019. DOI: 10.1088/1361-665X/ab3a77. [Online]. Available: https://dx.doi.org/10.1088/1361-665X/ab3a77.

[8] E. Hajiesmaili and D. R. Clarke, "Dielectric elastomer actuators," *Journal of Applied Physics*, vol. 129, no. 15, p. 151 102, Apr. 2021, ISSN: 0021-8979. DOI: 10.1063/5.0043959. eprint: https://pubs.aip.org/aip/jap/article-pdf/doi/10.1063/5.0043959/19887285/151102_1_5.0043959.pdf. [Online]. Available: https://doi.org/10.1063/5.0043959.

[9] X. Wang, S. K. Mitchell, E. H. Rumley, P. Rothemund, and C. Keplinger, "High-strain peano-hasel actuators," *Adv. Funct. Mater.*, vol. 30, no. 7, p. 1 908 821, 2020. DOI: https://doi.org/10.1002/adfm.201908821. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/adfm.201908821. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/adfm.201908821.

[10] S. Zhang, X. Ke, Q. Jiang, H. Ding, and Z. Wu, "Programmable and reprocessable multifunctional elastomeric sheets for soft origami robots," *Sci. Robot.*, vol. 6, no. 53,

eabd6107, 2021. DOI: 10.1126/scirobotics.abd6107. eprint: https://www.science.org/doi/pdf/10.1126/scirobotics.abd6107. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abd6107.

[11] T. Wang, H.-J. Joo, S. Song, W. Hu, C. Keplinger, and M. Sitti, "A versatile jellyfish-like robotic platform for effective underwater propulsion and manipulation," *Sci. Adv.*, vol. 9, no. 15, eadg0292, 2023. DOI: 10.1126/sciadv.adg0292. eprint: https://www.science.org/doi/pdf/10.1126/sciadv.adg0292. [Online]. Available: https://www.science.org/doi/abs/10.1126/sciadv.adg0292.

[12] S. Adam, L. Busoniu, and R. Babuska, "Experience replay for real-time reinforcement learning control," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 2, pp. 201–212, Mar. 2012, ISSN: 1558-2442. DOI: 10.1109/TSMCC.2011.2106494.

[13] M. Waltz and K. Fu, "A heuristic approach to reinforcement learning control systems," *IEEE Transactions on Automatic Control*, vol. 10, no. 4, pp. 390–398, Oct. 1965, ISSN: 1558-2523. DOI: 10.1109/TAC.1965.1098193.

[14] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, Nov. 2017, ISSN: 1558-0792. DOI: 10.1109/MSP.2017.2743240.

[15] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, *Proximal policy optimization algorithms*, 2017. arXiv: 1707.06347 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1707.06347.

[16] V. Francois-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, "An introduction to deep reinforcement learning," *Foundations and Trends® in Machine Learning*, vol. 11, no. 3, pp. 219–354, 2018, ISSN: 1935-8237, 1935-8245. DOI: 10.1561/2200000071. arXiv: 1811.12560[cs]. [Online]. Available: http://arxiv.org/abs/1811.12560 (Accessed 04/07/2025).

Appendix A Requirements and Verification

Table A.5: Requirements and verification table

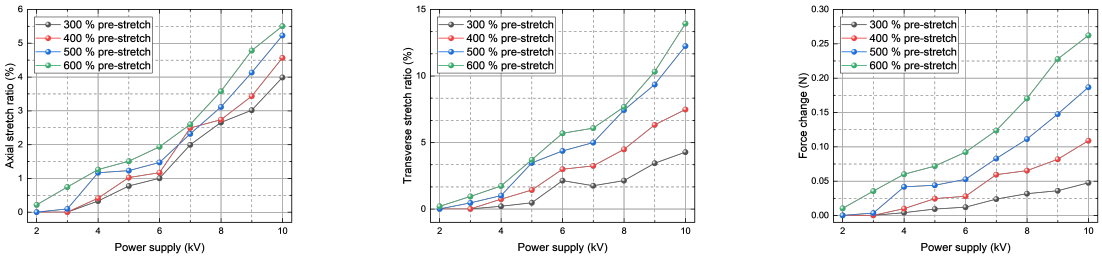
Requirements	Verification
1. Visual Processing Network	
1. Process data at 30 fps or higher. 2. Accuracy of recognition points must be above 95%.	1. A. Test at various frame rates to ensure 30 fps. B. Confirm accuracy greater than 95% via statistical sampling.
2. Computing Node	
1. Facilitate data exchange and computation. 2. Delay from input to output must be less than 200 ms.	1. A. Test peripherals to ensure proper functionality. B. Measure signal delay and ensure it is under 200 ms.
3. Reinforcement Learning Control Network	
1. Accomplish target reaching within 2 radii distance. 2. Follow planned path points reliably. 3. Achieve more than 30 fps control speed.	1. A. Validate task completion and distance. B. Ensure path-following with no obstacles. C. Benchmark control speed to meet 30 fps.
4. Motion Planning Program	
1. Calculate motion paths with 5+ points at more than 30 fps.	1. A. Verify path has more than 5 points and avoids obstacles. B. Test for robustness and speed of path planning.
5. Control Unit	
1. Total control time must be smaller than 150 ms. 2. Respond accurately to all commands.	1. A. Measure control time to meet the 150 ms requirement. B. Verify correct relay activation across all commands.
6. HV Amplifier	
1. Amplifier must provide sufficient gain and clear output.	1. A. Test gain and check for signal distortion.

Continued on next page

Table A.5 Continued from previous page

Requirements	Verification
7. HV Relay Set	
1. Ensure full charge/discharge of EBSA within control cycle. 2. Cost-effective and scalable for multiple units.	1. A. Measure EBSA charge and discharge times (<1 ms). B. Verify circuit operates with 2 relays per unit.
8. 2D-BJR (hydrogel electrode layer)	
1. Hydrogel must withstand deformation, retain conductivity, and maintain water content.	1. A. Perform tensile and fatigue tests. B. Measure conductivity and water retention.
9. 2D-BJR (main actuation structure)	
1. VHB films must form sealed pouch for electrohydraulic actuation. 2. Actuator must be prestretched for efficiency. 3. Actuator must form expected 3D shape after post-stretch release.	1. A. Inspect VHB pouch sealing. B. Test actuation at various prestretch levels. C. Confirm formation of jellyfish-like shape post-release.
10. Peripherals	
1. **IR signal Module** support signal transmission and reception with less than 100 ms delay. 2. **Camera** capture high-quality video at more than 30 FPS. 3. **Square Wave Generator** generates stable square waves at configurable frequencies. 4. **Glass Water Tank** be transparent and leak-proof to accommodate jellyfish model.	1. A. Test signal communication and transfer. B. Verify transmission/reception delay less than 100 ms. 2. A. Ensure stable video feed with no frame loss. 3. A. Verify waveform stability and frequency. 4. A. Check transparency and capacity. B. Test for leaks.

Appendix B Visual Results



(a) x'/x vs. voltage (b) y'/y vs. voltage (c) $F - F'$ vs. voltage

Figure B.12: Results of the pre-stretching experiment for linear DE-HASEL actuators.

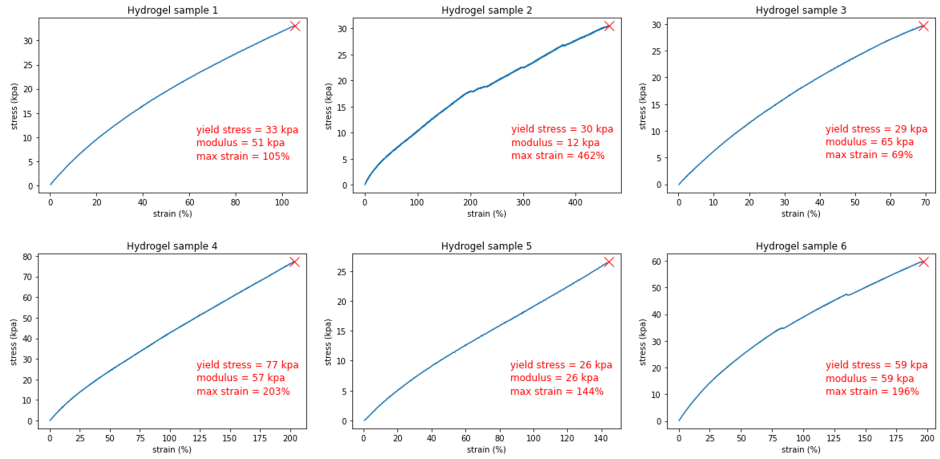


Figure B.13: Illustration of the strain-stress digrams for the hydrogel electrode layer.

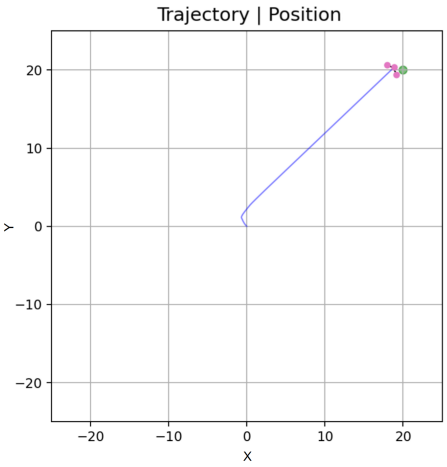


Figure B.14: Illustration of RL control process.

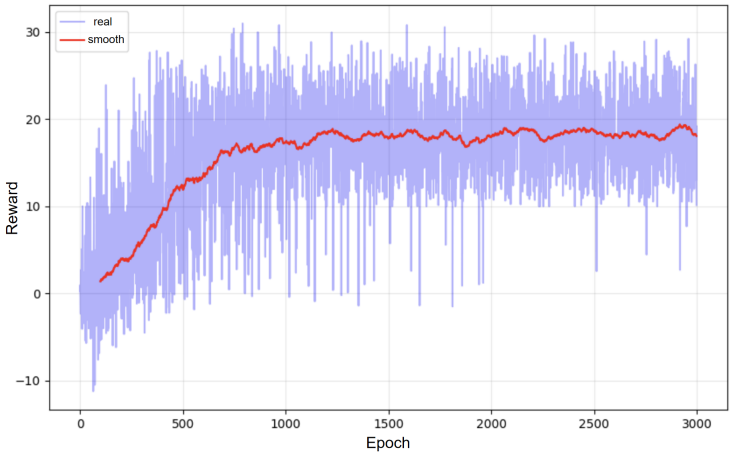


Figure B.15: Training curve of the RL agent.