Smart Agricultural Technology 12 (2025) 101168

Contents lists available at ScienceDirect

Smart Agricultural Technology

journal homepage: www.journals.elsevier.com/smart-agricultural-technology

Visual servoing of an underwater robotics arm for automatic sorting of crustaceans

Giacomo Picardi [a,*], Anna Astolfi [b], Karthik Seemakurthy [e], Bradley Hurst [d], Elena Piana [c], Petra Bosilj [d], Marcello Calisti [e,b]

[a] Instituto de Ciencias del Mar (ICM-CSIC), Passeig Maritim de la Barceloneta 37-49, Barcelona, 08003, Catalunya, Spain
[b] Sant'Anna School of Advanced Studies, The BioRobotics Institute, Pontedera, 56025, Tuscany, Italy
[c] Noola Redclaw ltd, Wimborne, BH21 7SF, Dorset, UK
[d] University of Lincoln, School of Engineering and Physical Sciences, Lincoln, LN6 7TS, Lincolnshire, UK
[e] University of Lincoln, School of Agri-Food Technology and Manufacturing, Lincoln, LN6 7TS, Lincolnshire, UK

ARTICLE INFO

Keywords:
Aquaculture
Robotics
User-driven design
Caging gripper
Computer vision
Crayfish

ABSTRACT

Crustaceans aquaculture is a rapidly growing sector, and automatising specific tasks can contribute to increasing its economic return as well as its sustainability. In this paper, we present an autonomous robotic system for sorting crustaceans by size, applied to crayfish farming. The system was designed in close collaboration with a crayfish farming company following a systematic user-driven approach. It consists of a waterproof robotic arm with a custom caging gripper lodging a camera, and a vision system to detect crayfish and sort them by size. All aspects of system design are presented: from manipulator and gripper design and control, to the development of the vision system, and the system integration. The system is evaluated in a controlled laboratory environment using synthetic crayfish models, and in a tank with live crayfish on a farm. Our evaluation shows that the presented system is capable of recognizing and selecting crayfish based on their size, and safely entrapping them in the caging gripper without causing any damage.

1. Introduction

Crustaceans are a crucial food source for much of the global population, and their aquaculture presents a sustainable means to meet worldwide demand. In 2020, according to the Food and Agriculture Organization of the United Nations (FAO), total aquaculture production of crustaceans reached 11.2 million tons, valued at an estimated 81.5 billion US dollars, surpassing production from capture fisheries [1].

Sorting and monitoring the growth of certain crustacean species, such as crayfish, lobster, and crabs, are essential for successful farming. These processes improve product quality, reduce injuries and mortality from aggressive behaviour, and increase productivity and profitability for farmers. However, these tasks are labor-intensive and often rely on temporary workers, who are increasingly difficult to find. Automation offers a promising solution to significantly enhance crustacean farming productivity. Nonetheless, various challenges remain regarding the specific processes that can be automated and the available technologies.

A recent review by Li et al. [2] summarized the state of the art in automatic monitoring of relevant crustacean behaviours in aquaculture, categorizing these technologies as acoustic, computer vision, or sensor-based systems, such as accelerometers and electromyography. Reis et al. [3], among others, have focused on optimizing automated feeding in shrimp aquaculture and defining standardized feeding protocols to maximize growth rates in semi-intensive shrimp ponds. Other researchers and companies have developed comprehensive solutions, such as a commercial land-based system for the farming of European lobsters and other crustaceans that incorporates automation for accurate feeding, juvenile mass-rearing, and image-processing programs for daily monitoring of individual specimens [4,5]. Certain crustacean species, like crayfish, are also farmed as bioindicators. For example, Kuklina et al. [6] provided an in-depth review of studies using crayfish to assess

* Corresponding author.
E-mail addresses: giacomo.picardi1991@gmail.com, gpicardi@icm.csic.es (G. Picardi).

https://doi.org/10.1016/j.atech.2025.101168
Received 1 April 2025; Accepted 3 July 2025
Available online 8 July 2025

water quality. Similarly, LeCocq et al. [7] developed a system to identify male adult amphipods used in bioconcentration studies, employing an image-based algorithm for sexing and a robotic arm with a suction syringe to separate males from females.

In general, most efforts to automate crustacean aquaculture have addressed tasks like feeding, tank cleaning, or assessing health and growth. However, to the best of the authors' knowledge, no existing work addresses handling live, non-anaesthetized specimens (e.g., catching and sorting). This paper presents an automated crustacean sorting system (RoSC) designed to automatically scan the tank, detect live, non-anaesthetized crayfish of a specified size, catch them without causing harm or stress, and remove them from the tank. We followed a systematic user-driven design approach developed in close collaboration with a crayfish farming company (Noola RedClaw Ltd). This allowed us to incorporate end-user requirements from the early design stages and gain access to aquaculture facilities for preliminary data collection and real-condition validation. The final system comprises a waterproof robotic arm, a cage-like gripper with an embedded camera, and a vision system to detect and classify individuals by size. In the remainder of this paper, we present the design and development of the system, along with its validation through experiments conducted on synthetic crayfish in the lab and live, non-anaesthetized crayfish on the farm. We discuss the final design and experimental results, highlighting potential improvements and the advantages of a systematic user-driven approach in agricultural robotics applications.

2. Related works

2.1. Waterproof robotic arm and gripper designs

As noted above, there are currently no reports in the literature of robotic systems capable of capturing live, non-anaesthetized fish or crustaceans without causing harm [8]. The need for waterproof hardware is crucial in such applications. Commercially available depth-rated manipulators are costly, while custom-designed manipulators are more prone to leaks, which can result in significant time losses or even permanent damage. Examples of robust underwater manipulator designs can be found in this recent review [9] or in the literature on Underwater Legged Robotics (ULR), in which several custom-designed legs have been developed [10]. Furthermore, many fish and crustaceans, including crayfish, have the unique ability to perform rapid full-body escape manoeuvres [11], making it difficult to track and capture them using conventional robotic approaches. For this reason, a deep understanding of animal behaviour and the best practices adopted by farms is crucial to developing the most effective control strategy.

In field studies, marine biologists often rely on traps to capture crayfish [12]. However, these methods are unsuitable for targeting specific individuals, as required in our work. Alternatively, crayfish aquaculture workers use dip nets, slowly lowering the net over the animal and then grasping it by hand. Although this technique has been effective over years of practice, replicating it in a robotic system would require two manipulators with different end-effectors, thereby increasing both cost and control complexity. Recent multidisciplinary collaboration between roboticists and marine scientists has led to the development of grippers tailored for sampling different types of marine animals [13]. Caging grippers, in particular, resemble dip nets, but feature ribs along the net that can be activated to close around and contain the animal with minimal contact. This design eliminates the need for a second manipulator and ensures safe interaction with the animal.

2.2. Computer vision detection architectures for crustaceans

Especially since the advent of deep learning, computer vision approaches have seen many applications in aquaculture [14]. However, applications involving live crustaceans remain limited. Hasan et al. [15] proposed a template matching approach for identifying species, age and sex of lobsters, however requires close-up images taken outside of the water and is therefore not suitable for live crayfish sorting. Wang et al. [16] have deployed a lightweight SSD detector with MobileNet V2 backbone onto a spherical amphibious robot to detect the shrimps, however the reported inference speed (19 seconds per frame) is not sufficient for tracking crayfish. Vo et al. [17,18] propose a system for grading and identification of lobsters, however also require close-up images taken under controlled conditions. Ye et al. [19] use YOLOv5 for classifying crayfish into 8 classes according to size and maturity. While their work is most similar to ours, all their images were taken on a conveyor belt from a fixed height where crayfish were facing approximately forward, and additionally the size was not measured exactly but rather sorted into small, medium, large and extra large. While both our work and [19] use a model based on YOLOv5 [20], our work additionally estimates the orientation of the crayfish using oriented bounding boxes [21] to obtain an exact measure of crayfish size.

2.3. Systematic design process

Robotics is moving from structured industrial settings to more diverse and challenging environments that are difficult to predict, and agricultural robotics is leading the trend due to sustainability and economical factors [22]. However, surveys highlighted end-user are rarely taken into account and real-world impact is relatively poor [23]. To the author knowledge, a structured attempt to develop new robotic solutions for the agri-food sector does not exist, which most authors gaining qualitatively insights to develop or adapt existing robots [24], or taking only in partial consideration the entire pipeline or the farmers' insights.

For example, [25] designed a radicchio harvesting end effector based on practical harvesting requirements (i.e. penetration about 10 cm below the ground) while [26] optimized the geometry of a manipulator to exploit the optimal path for cucumber harvesting. On a more conceptual level, system requirements have been used to develop a teleoperated robotic system for watermelon [27], without explaining the methods used to derive such specifications, and eventually pick-and-place operations have been split down into sub-tasks to optimize a multi-purpose food handling robot [28]. In all such cases, feedback and insights from farmers are not collected in a structured way and it is difficult to understand how they are exploited and incorporated into the final design.

In our work we followed a Quality Function Development approach, through the construction of a House of Quality, that allows a clear and time-saving approach to incorporate farmers' and developers requirements, identify engineering specifications, and to built our solution upon existing ones.

3. Design and construction of robotic arm and gripper

3.1. User-driven design of robotic arm and gripper

The main goal of the RoSC project was to develop a robotic system capable of sorting crayfish by size while they are freely moving within their tanks and capturing them without causing harm or stress. From the preliminary discussion with the farming company Noola RedClaw Ltd and multiple visits to the aquaculture facilities, several, sometimes conflicting requirements emerged. To address such requirements systematically and guide the design process, we applied the approach detailed in [29] which can summarized as follows. A House of Quality (HoQ) framework was used to capture the primary requirements of various stakeholders and transform them into measurable engineering specifications. State-of-the-art solutions were evaluated against these specifications to guide the conceptualization of the novel design. The full HoQ for the RoSC system can be found in the supplementary material. The highest-priority requirements included "waterproofing", "lifting animals of various sizes", "avoiding harm or distress to the animals", along with "ease of assembly", "transportability", "cost-effectiveness", and "adaptability to different farms and growth conditions". From these,

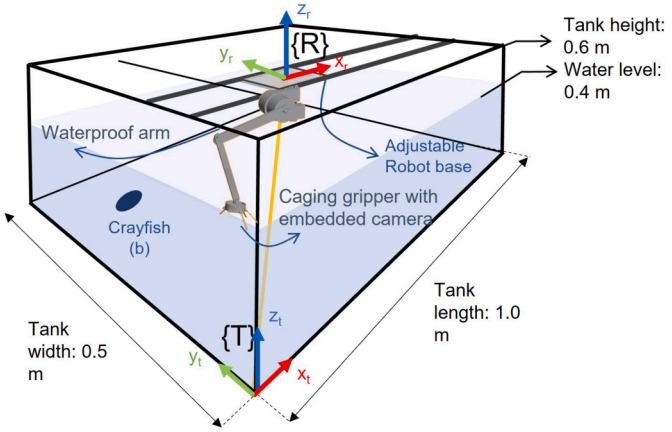


Fig. 1. Conceptual design of the presented system. A robotic manipulator with 3 degrees of freedom, attached to an adjustable robot base, is installed on top of an aquaculture tank. The end effector of the manipulator is a caging gripper designed to safely trap and move crayfish outside the tank. A camera is embedded in the palm of the gripper to enable image-based detection, classification, and tracking of crayfish. In this depiction, the net of the caging gripper and the camera are not shown.

Table 1
The most relevant engineering specifications derived with the proposed design methodology for the presented RoSC system and for the leg of SILVER2.

	Description	Goal	SILVER2	RoSC
ES1	Number of interfaces	↓	12	3
ES2	Stiffness of compliant elements	↑	660 N/m	NA
ES3	Number of waterproof components	↓	12	4
ES4	Actuators torque	↑	4.1 N/m	9.1 N/m
ES5	Number of actuators	↓	3	4
ES6	Number of sensors	↓	3	2
ES7	Number of assembling tools	↓	7	2
ES8	Number of components	↓	>120	74

we derived a set of engineering specifications (ES), which guided development and served as criteria for final system validation. The most relevant ES are reported in Table 1

Two systems of the state-of-the-art were evaluated against these specifications, representing opposite approaches to the problem. A fully integrated patented solution [5], and a waterproof manipulator with three degrees of freedom (DoFs) electrically actuated as the leg of the Underwater Legged Robot SILVER2 [30]. Eventually, the importance given by the farming company to requirements such as 'easy assembly', 'transportability', 'cost-effectiveness' resulted in [30] scoring higher and being taken as a reference for the current design. The conceptual design of the RoSC system is shown in Fig. 1. It includes a custom-designed waterproof robotic arm (manipulator) with three DoFs, mounted on an adjustable base, and equipped with a soft robotics gripper. The arm is fully waterproof and can be suspended from any suitable supporting structure, enabling adaptability to different farm setups and a fully submerged installation. In our prototype, the arm was suspended from two horizontal bars to meet to adapt to different tank sizes. The manipulator's design keeps most of the weight near the adjustable base. With respect to the solution presented in [30], in which each motor is individually sealed, we developed a custom canister that houses three motors together and employs a bar transmission system to actuate the most distal joint. This approach granted a reduction of the manipulator's inertia and hydrodynamic drag and contributed to overall simplification of the hardware.

The manipulator's end-effector is a soft caging gripper [13] with an embedded camera in its palm. This gripper type was selected based on the system requirements, existing underwater sampling methods for remotely operated vehicles (ROVs), and preliminary behavioural observations in the farm. While observing crayfish sorting procedures on-site, we noticed a consistent escape response when crayfish were approached from the side, typically as the fishing net dragged along the tank floor to trap them. In contrast, as shown in Fig. 2, crayfish did not react to a mock-up gripper approaching from above, allowing them to be easily trapped. This observation influenced both the gripper design and arm control strategy. The gripper is connected to the manipulator via a passive pin joint, enabling it to face downward regardless of the last link's orientation, thereby eliminating the need for an active wrist. We embedded a single camera in the gripper palm, keeping the system self-contained and minimizing the number of sensors.

The system was designed for a 1.0 m x 1.0 m x 0.6 m tank where crayfish of various sizes can move freely. Link dimensions were optimized to maximize coverage of the tank floor within the manipulator's workspace; details of these calculations are provided in Section 4.

3.2. Construction of robotic arm and gripper

The RoSC manipulator has three degrees of freedom, and it is composed of three main subsystems: a planar four-bar linkage, a shoulder mechanism, and a caging gripper (see Fig. 3a). When the gripper is detached, both the red and green links of the four-bar linkage can complete a full rotation, classifying it as a double-crank mechanism with equal link lengths (Fig. 3b, dashed square). The actuated crank (red) reaches a singular position at 0 or π positions (fully folded or extended); software and mechanical constraints prevent operations in these positions. At the end of the green link, the caging gripper is passively mounted via a rotational joint.

The shoulder mechanism is custom-designed to meet critical waterproofing requirements. The shoulder has been engineered to concentrate the manipulator's weight near the base, reducing inertia, and to limit penetrations by using a single canister for all three motors. This two-part cylindrical canister is sealed with a front o-ring and a single gland penetrator for power and communication lines (static penetrations), while four o-rings seal the rotating elements (Fig. 3b). Motor m_1 rotates the entire manipulator, while motors m_2 and m_3 actuate the four-bar mechanism. Both motors m_1 and m_2 are connected directly to one side of the canister, minimizing the number of penetrators required for chains of smart servomotors.

To drive the four-bar mechanism, motor m_2 (Fig. 3c) simultaneously rotates both the blue linkage and motor m_3, via the rigidly connected components c_1 and c_2. Component c_2 rotates against the canister and is sealed with an o-ring. Motor m_3 then rotates the yellow component c_3, which is attached to the red crank, completing the four-bar actuation. Component c_3 rotates against c_2, insulated with an additional o-ring. Notably, the canister also houses a fourth motor, reserved for future use to control the gripper, as shown in Fig. 3b.

The arm is completed by the caging gripper, depicted in Fig. 3d. This gripper consists of six radial beams (or "fingers") extending from a central palm structure. Connecting features allow a plastic net to be threaded around the beams, forming a fully enclosed cage. Within the palm, a camera is embedded and waterproofed with epoxy resin encapsulation. The epoxy resin's weight beneath the rotational pin passively keeps the palm horizontal, which is optimal for caging and approaching crayfish.

The modular RoSC design enables independent work on each subsystem, simplifying maintenance and providing easy access. Fabricated components from the assembly and manufacturing processes are shown in Fig. 4. The four-bar mechanism was CNC-milled from white Acrylonitrile Butadiene Styrene (ABS), while the canister, shafts, and connecting parts were produced with stereolithographic printing in ABS-like resin (Protolabs). The gripper structure was 3D printed using a Formlab Form3 resin printer with Rigid 10 K resin, washed, and cured following Formlab instructions. The fingers were printed with the same printer, using Tough 1500 resin, also washed and cured as per Formlab guide-

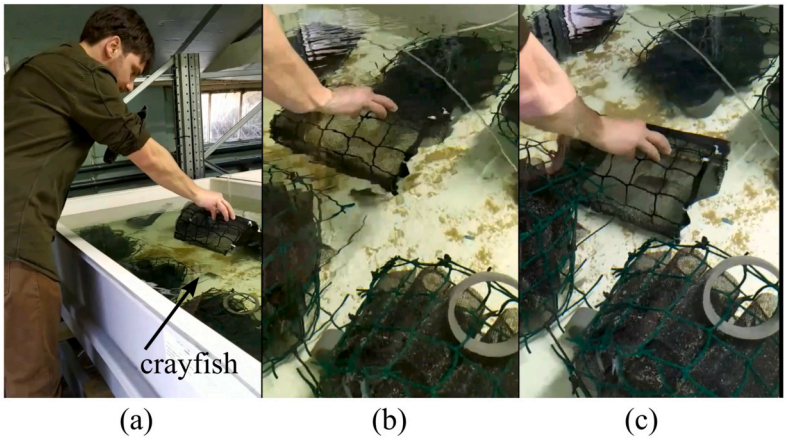


Fig. 2. Trapping experiments with a mock-up caging gripper on live non-anaesthetized crayfish in their tanks. Slowly approaching the crayfish from above does not trigger their escape response.

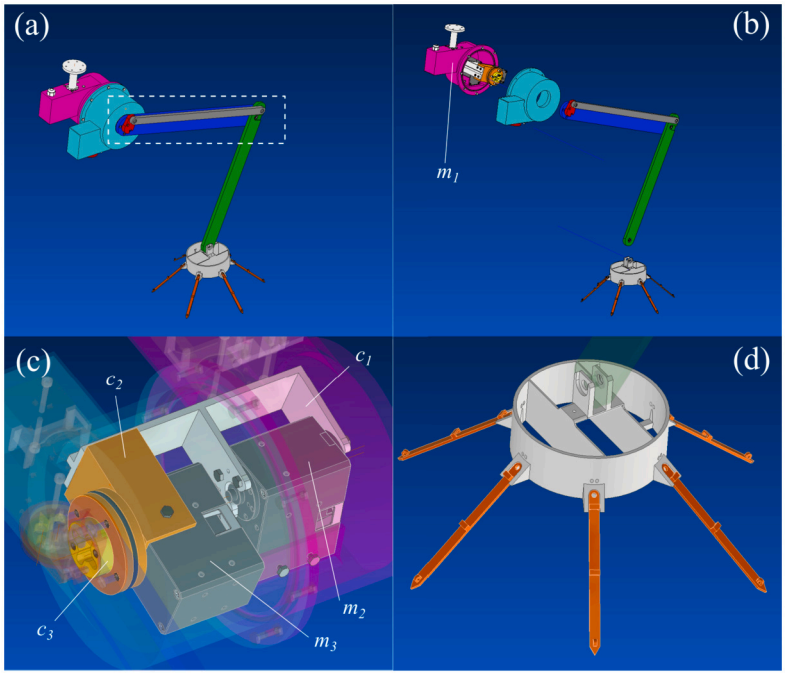


Fig. 3. (a) Overview of the RoSC manipulator including the planar four-bar linkage, a shoulder mechanism, and a caging gripper. (b) Exploded view of the RoSC manipulator. (c) Detail of the shoulder mechanism. (d) Caging gripper (net not shown).

lines. A Logitech camera (model to be specified) was encapsulated in clear epoxy resin. The arm's actuators are Dynamixel XH540-W270-R servo motors, and the entire robotic system is controlled by a Raspberry Pi4 board.

In terms of the ES defined in section 2.3, the present implementation improves the reference design [30] for almost all the considered fields. In particular, the number of interfaces (ES1) was reduced from 4 cables and 9 penetrators to 2 cables and one penetrator, thanks to the single sealed canister solution. The compliant element which was present in [30] was removed, and the compliance was concentrated in the ribs of the gripper (ES2). The number of waterproofing components (ES3) was reduced to 3 dynamic sealings for the motor shafts and 1 penetrator. Actuators with higher torques (ES4) were selected almost doubling the torque of the actuators in [30], while the adoption of an underactuated gripper and a passive ball joint between the manipulator and the gripper granted a minimum number of 4 actuators (ES5). In this case [30] presents a lower number of actuators with respect to the present design, but it lacks the end-effector. The low number of sensors (ES6) is obtained harnessing the compliance of the gripper for delicate grasping, using a single camera embedded in the palm, and reducing the number of leakage sensors to one, as a consequence of the single canister solution. Finally, the simplicity of assembly is demonstrated by the drastic reduction of assembling tools required (ES7) and overall number of components (ES8). The total time to assemble the system is below 15 minutes.

4. Modelling and control

This section addresses the modelling and control of the system. We chose to control the manipulator's joints in position, utilizing the low-level control SDK provided by the motor manufacturer. In the first subsection, we derive the manipulator's kinematic model, which serves as the foundation for the controllers used in various subtasks. The second subsection introduces the finite state machine that coordinates these

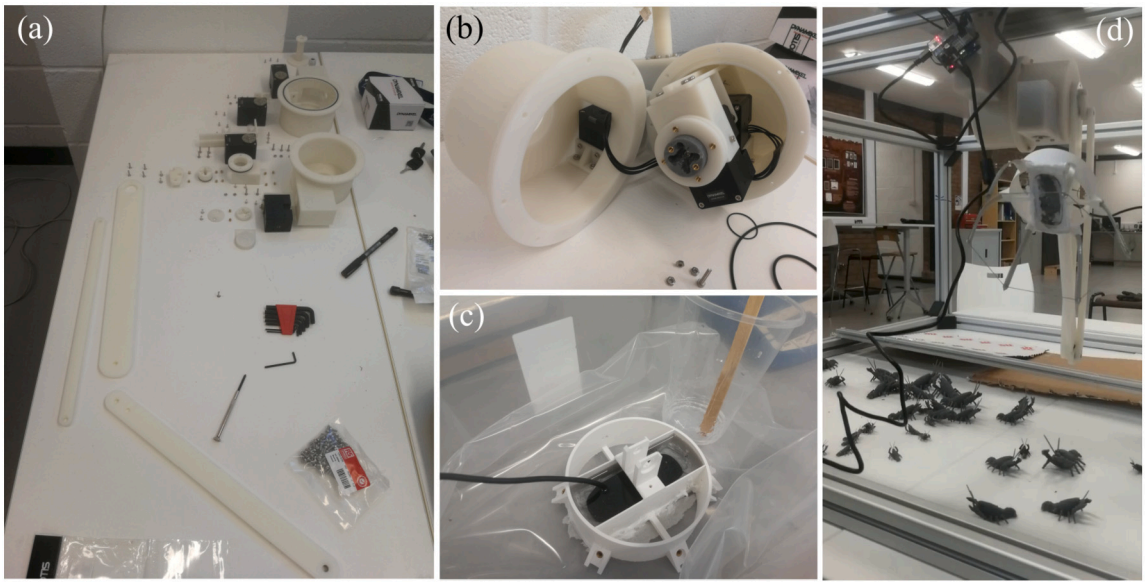


Fig. 4. (a) shows all the components of the arm (gripper excluded); (b) shows the assembly of the cannister; (c) the resin encapsulation of the camera in the gripper; and (d) the complete system attached to the supporting bars.

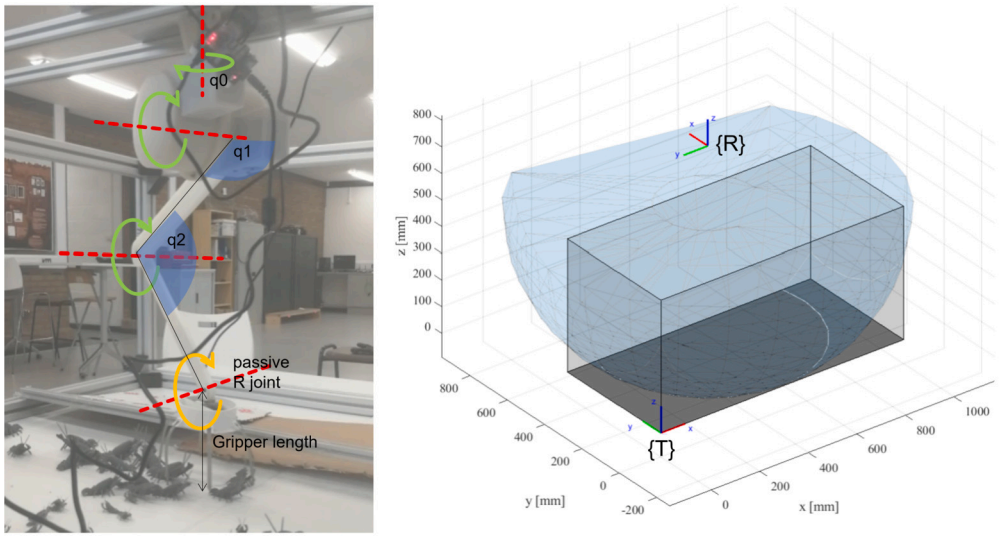


Fig. 5. (a) RoSC system assembled with configuration angles shown. (b) Work space of the end-effector position obtained from the manipulator kinematic model.

Table 2
Link lengths of the RoSC manipulator.

Links	l_0	l_1	l_2	l_3	l_4	l_5	l_6
Measure (mm)	103.26	60.75	106	350	12.5	300	107

controllers based on the robot's current state, along with further details on each controller.

4.1. Manipulator kinematics and workspace evaluation

The kinematic model of the robotic arm is depicted in Fig. 5. The robot features 4 controllable DoFs, and 1 uncontrolled DoF. The first three DoFs are used for controlling the configuration of the arm, whereas the last DoF is used for opening and closing the caging gripper. The uncontrolled DoF corresponds to the passive spherical joint connecting the gripper to the manipulator, and it allows the gripper to face downward irrespective of the orientation of the last link of the robotic arm. The robot home configuration (all joints at 0 position from shoulder to wrist) is defined by the following matrix:

$$M = \begin{bmatrix} 0 & 1 & 0 & l_2 + l_4 \\ 0 & 0 & 1 & l_1 \\ 1 & 0 & 0 & l_5 - l_0 - l_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

with the parameters $l_{0\ldots6}$ as in Table 2. The forward kinematics can be computed using the approach presented in [13], starting from the motor angles θ and screw axis S as the product of matrix exponentials:

$$T_{ee} = \Pi_{i=0}^{2}(e^{[S_i]\theta_i})M \tag{2}$$

where $[S]$ is the $se(3)$ matrix representation of the $6x1$ screw axis S. The position of the end-effector, defined as the centre of the caging gripper, can be obtained by offsetting the z position of the wrist.
Given these notions, the robot workspace can be computed, and it is represented in Fig. 5, together with the tank workspace, corresponding to the physical space that the system should reach. The inverse kinematics was solved numerically using the Newton-Raphson method.

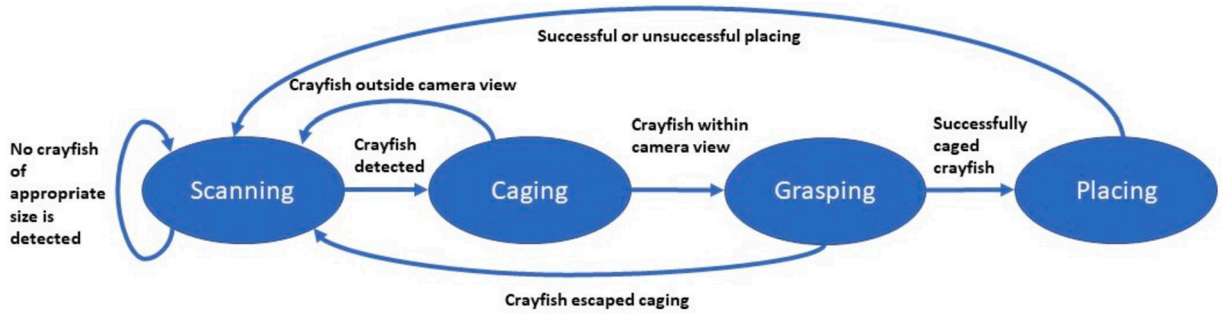


Fig. 6. Depiction of the Finite State Machine describing the task planner for the proposed solution.

4.2. Task planner

Automating the sorting of crayfish with the hardware described in section 3.1 was divided into four subtasks, corresponding to the states of the Finite State Machine (FSM) shown in Fig. 6. These four states—Scanning, Caging, Grasping, and Placing—are outlined below. In this work, we focused on the implementation and testing of the Scanning and Reaching states, which, according to crayfish farming professionals, represent the toughest challenges due to the risk of startling the animals or damaging them during robotic operations.

1. **Scanning**. This is the state in which the system enters upon starting. The manipulator follows a pre-defined scanning path, keeping the gripper at a fixed distance from the bottom of the tank while running the detection model described in Section 5. In case no crayfish of appropriate size is detected, the scanning task is started over again. Otherwise, the manipulator is stopped, and the system switches to the Reaching-state. The scanning path within the unsorted area of the tank is depicted in Fig. 7.
2. **Caging**. The goal of this state is to bring the caging gripper on top of the selected crayfish to prevent it from escaping. The position of the centroid of the bounding box of the crayfish in the camera frame C is tracked, and used as goal position p_{des} in a closed loop point reaching task. The control law implemented for this task is a Proportional-Derivative controller (PD). The feedback signal is the joint position error $e_q = q_{des} - q_{cur}$, where q_{des} is obtained by converting p_{des} from C to the robot reference frame R (Fig. 7) and applying the inverse kinematics, and q_{cur} is obtained by applying the inverse kinematics to the manipulator current position p_{cur}. In case the selected crayfish moves outside the field of view, the FSM goes back to the Scanning-state. Otherwise, the system stops when the goal position is reached and the crayfish is trapped under the caging gripper. In case of successful reaching, the FSM moves onto the Caging-state.
3. **Grasping**. In the Grasping-state the goal is to activate the caging gripper to safely trap the selected crayfish. In case the crayfish escapes the caging, the FSM goes back to Scanning-state, otherwise it moves onto the Placing-state. The success of the Caging-task can be simply assessed through the camera feedback.
4. **Placing**. The goal of this state is to transfer the crayfish into the sorted area. The manipulator is controlled to reach a predetermined point within the unsorted area and activate the gripper to open it and release the animal. Regardless of the outcome of this task, the FSM goes back to the Scanning-state.

5. Vision system development

Two datasets were collected and annotated for the development of the vision system, one on synthetic 3D printed crayfish for preliminary testing, and one live crayfish. Firstly, as access to live crayfish was limited, 145 s of video was recorded using RedMi Note 4 mobile phone

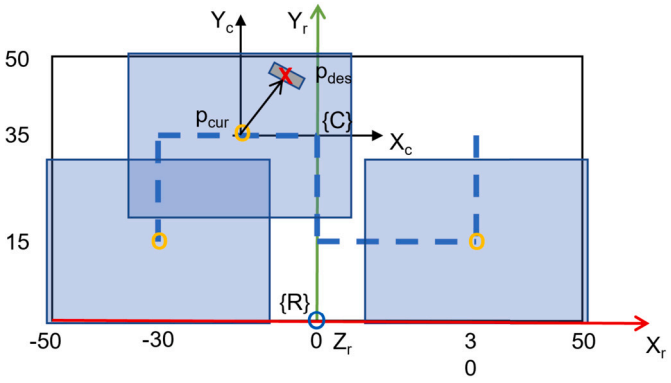


Fig. 7. Schematics of the unsorted area of the tank. The dashed blue line represents the scanning path which the end effector follows during the Scanning-task. The blue rectangles represent the area observed by the camera when the end-effector is on the yellow circle. In the Reaching-task, the desired position p_{des} is the centroid of the bounding box of the crayfish, whereas the current position of the end-effector is identified by p_{cur}.

camera 3D of printed model crayfish in three different sizes S1 = 3.5 cm, S2 = 6.5 cm and S3 = 10 cm, and resulted in 512 annotated images of synthetic crayfish taken from varying heights, to simulate the movement of the robotic arm. Later, 449 s (720x1280 pixel at 30fps) of video were recorded using a PELLOR water proof camera, separately from water tanks with large and small specimen weighting around 8 and 3 grams respectively. After curation, 722 images were annotated using the LabelMe [31] tool, with each crayfish denoted by an oriented bounding box (OOB) taking only a few seconds to annotate. Out of these, the majority (583 images) were taken from a fixed height of 40 cm, selected as the default height of the gripper to obtain adequate field of view, with only a small number (139 images) taken from varying camera heights.

Considering the need to estimate the length of each crayfish, we considered a single-stage object detection approach modified to work with OOB YOLOv5-OBB [21] and an instance segmentation approach Mask-RCNN [32]. Based on the inference times on Raspberry Pi 4, we discarded Mask-RCNN (0.5fps) and used YOLOv5-OBB (5-10fps) as a basis for our vision model. YOLOv5-OBB is based on YOLOv5 [20], and regresses orientation in addition to the bounding box size and objectness score for each image position. As the gripper will be at a fixed height during detection and before initialising the grasping operation, the length of the crayfish was determined by multiplying the length of the crayfish in pixels by the pixel resolution, determined through calibration. We used the Darknet-53 [33] backbone, initialised with COCO-pretrained weights [34] as our encoder.

Mean average precision (mAP) at 50% Intersection over Union (IoU) - mAP@50 is selected as the evaluation metric [35] Two models were trained, each for 100 epochs using the Adam optimiser [36], both final models initialised with a learning rate of 0.01. First, the synthetic crayfish dataset was used (divided into training/validation/test sets of size 473/20/20 images) to select the best learning rate and image augmen-

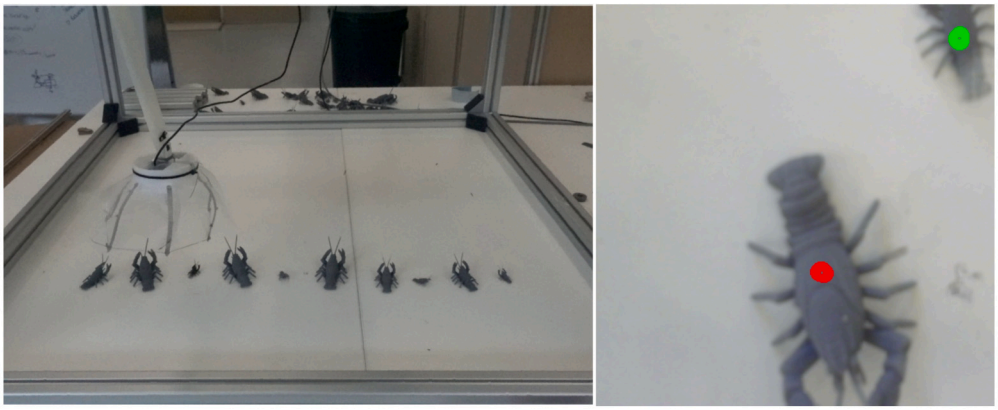


Fig. 8. Laboratory setting for the experimental trials. Crayfishes detected but outside the desired range of size were marked with a red circle; crayfishes of the desired size were marked with a green circle.

tations capable of achieving the highest performance in light of scale changes, with the best model achieving 0.988 mAP on the synthetic test set. Secondly, the model for the vision system was trained using the real crayfish dataset (divided into 566, 153 and 53 images in the training, validation and test sets respectively), while applying median blur (with the probability of 0.5 and aperture linear size (3,7) and random cropping (with the probability of 0.25 and crop size varying between (50x50) and (100x100) pixels) during training to increase the visual diversity of the dataset, and achieving 0.957 mAP on the test set with real crayfish, which is in line with performance reported by related works [19]. However, during the final test in a real crayfish tank with real live crayfish, it was found that the model did not initially detect any crayfish. Upon inspection of the model outputs, only the threshold for accepting a detection was lowered from the default value of 0.5 to 0.1, while leaving all other model parameters unchanged, to finalise the vision system.

6. System integration and experimental procedure

The manipulator control and the vision system were integrated using a ROS-based architecture. A manipulator control node implemented the FSM and the Scanning and Caging behaviours presented in 4.2. The input for the transition between Scanning and Caging states was provided by a vision node implementing the detection model presented in 5 and publishing both the pixel-wise x, and y coordinates of the detected crawfish, and a masked version of the sampled frame. OpenCV was used to produce the mask which highlights the centroids of the detected crawfish in green for specimen of the desired size and red for those estimated to be too small or large, as depicted in Fig. 8. The Open Neural Network Exchange (ONNX) library was used to integrate the model into the node. The experimental setup and procedures used to evaluate the RoSC system both in a lab environment with artificial 3d printed crayfishes, and within the actual tank in the farm environment with live, non-anaesthetized animals, are detailed in the remainder of this section.

6.1. Lab and synthetic model experiments

Prior to experiments in the farm, the RoSC system was tested using 3D printed synthetic crayfish in the lab. The goal of these experiments was to validate the mechanical design and test the integration of the various subsystems. In the farming industry, crayfish are sorted into three categories with small crayfish presenting body lengths up to 5 cm, medium crayfish between 5 and 7.5 cm, and large crayfish between 7.5 and 11 cm. According to this classification, synthetic crayfishes were 3D printed with three different sizes: $S_1 = 3.5$ cm, $S_2 = 6.5$ cm and $S_3 = 10$ cm, to fall within the sorting ranges suggested by the farmer. The experimental procedure conducted in the lab comprised randomly selecting a pool of eight to ten synthetic crayfish, and placing them in a line as depicted in Fig. 8. The RoSC arm, mounted on a metallic frame, scanned the line from one side, and lowered the gripper to cage crayfishes of a specific size upon detection. When a synthetic crayfish is caged by the system, it is manually removed from the line and the scanning continues. A trial is considered completed once the whole line had been scanned back and forth. For each of the three pools of synthetic crayfish, one target size was selected, and a total of ten trials were performed. For each trial, the number of False Positives, i.e. the attempts to cage a synthetic crayfish of a different size, and False Negatives, i.e. do not attempt caging a synthetic crayfish of the correct size, were recorded.

6.2. Farm experiments

The testing protocol in the farm aimed at further validating the mechanical system, testing the vision system on real crayfish, tuning of the controllers parameters, and a final evaluation of the integrated system. Four tests were carried out as described below. In the first two tests, the vision system was off, and the focus was on the selection of the arm velocity. In the third test, the size estimation capability of the vision system was evaluated without triggering the caging behaviour. Finally, in the last test, the loop between the arm control and the vision system was closed, and the experiments were carried out autonomously. In all the tests, the robot was instructed to scan the tank with the grid trajectory reported in Fig. 7.

- **Test 1.** The arm scanned the tank and one operator manually triggered the caging controller. The gripper was kept down on the tank bottom for $4s$ and then the scanning trajectory was manually resumed. A total of 12 caging attempts were performed. The velocity of the arm during both Scanning and Caging was set to low (i.e. $5cm/s$).
- **Test 2.** The same procedure of Test 1 was adopted with a higher velocity of the arm during both Scanning and Caging behaviours (i.e. $15cm/s$).
- **Test 3.** The arm scanned the tank with the detection node and size estimation running. For each detected crayfish, we compared the actual length with the output of the vision system. In this test, caging was never attempted. The velocity of the arm during scanning was set to low (i.e. $5cm/s$).
- **Test 4.** The arm scanned the tank and automatically invoked the caging controller upon the detection of crayfish of the desired size by the detection node. The scanning trajectory was automatically resumed after $4s$. The velocity of the arm during both Scanning and Caging was set to slow (i.e. $5cm/s$).

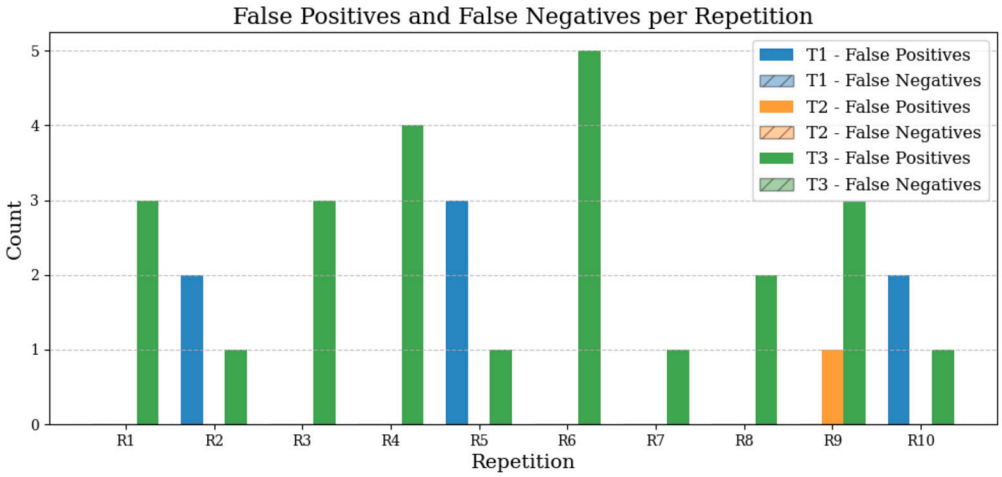


Fig. 9. Number of False Positives recorded over ten trials for the three pools of synthetic crayfish.

Table 3
Composition of synthetic crayfish pools used in the lab trials.

	S1	S2	S3	Tot
P1	3	3	4	10
P2	4	3	3	10
P3	3	2	3	8

Table 4
Confusion matrix related to P1.

T1. Target Medium	Actually Positive	Actually Negative
Predicted Positive	**TP 100%** (30/30)	**FP 10%** (7/70)
Predicted Negative	**FN 0%** (0/30)	**TN 90%** (63/70)

Table 5
Confusion matrix related to P2.

T2. Target Small	Actually Positive	Actually Negative
Predicted Positive	**TP 100%** (30/30)	**FP 1.4%** (1/70)
Predicted Negative	**FN 0%** (0/30)	**TN 98.6%** (69/70)

Table 6
Confusion matrix related to P3.

P3. Target Large	Actually Positive	Actually Negative
Predicted Positive	**TP 100%** (30/30)	**FP 48%** (24/50)
Predicted Negative	**FN 0%** (0/30)	**TN 52%** (26/50)

Table 7
Confusion matrix for crayfish detection during test 4 in the farm. A total of 4242 samples were analysed. Size is ignored.

Test 4	Actually Positive	Actually Negative
Predicted Positive	**TP 2%** (49/464)	**FP 10%** (63/3778)
Predicted Negative	**FN 98%** (415/464)	**TN 90%** (3715/3778)

7. Results

7.1. Synthetic crayfish experiments

For each trial (T1-T3) a corresponding pool of synthetic crayfish was randomly sorted (P1-P3) as reported in Table 3. In T1 the system targeted medium-sized crayfish, in T2 the small-sized ones, and in T3 the large-sized ones. No FN were recorded during this experiment, and the complete summary of the trials is shown in Fig. 9. The absence of FNs means that the system always attempted caging a synthetic crayfish of the correct size. On the other hand, the presence of FPs indicates that the system sometimes attempted caging a synthetic crayfish of the wrong size. By looking at the composition of each pools of synthetic crayfish (Table 3) and aggregating the count of FP and FN across the ten trials, it is possible to generate one confusion matrix per target size. For each trial, the resulting confusion matrixes are reported in Table 4, 5, and 6. The lower number of errors was recorded in T2, when small-sized crayfish were targeted. On the other hand, the higher number of errors was recorded in T3, when large-sized crayfish were targeted.

7.2. Farm experiments

The fully integrated system during farm experiments is depicted in Fig. 10. As described in section 6.2, tests 1 and 2 followed the same protocol, with the only difference being the arm velocity. In both cases, the arm and gripper hardware and low level control were proven reliable, as no failure was observed. When the arm velocity was set to slow (test 1), we counted a total of 10/12 successful caging (83.33%), whereas for higher arm velocity (test 2) we counted 9/12 (75.00%) successful caging. After caging, during test 1, one crayfish managed to escape, resorting to their fast escaping mechanism and harnessing a small aperture in the gripper cause by a misplacement of the net, whereas in test 2, no crayfish was capable of escaping from the gripper. The arm velocity for the following tests was set to slow (5 cm/s) as a consequence of the higher number of successful caging observed in test 1 with respect to test 2.

Test 3 dealt with the evaluation of the size estimation capability of the vision system. The results of this rest are summarized in Fig. 11 in which the erroneous detections were neglected. It is clear that the vision system consistently underestimate the crayfish length, with an average relative error of 31.43%. The reason for such consistent underestimation is due to the curved tail of crayfish. This result explains why during lab trials, the higher number of FP was observed for P3, i.e. while looking for large crayfish.

In test 4 we aimed to close the loop between the arm control and the vision system. The size detection was ignored in this test, thus all sizes were considered valid targets. During the test, the RoSC system ran for approximately 170 minutes at 25 fps, collecting a total of 4242 frames. In Fig. 12, four examples of frames acquired with the onboard camera and output of the vision system are reported. TPs, as shown in Fig. 12a, are frames containing a crayfish correctly identified and marked by a green dot. FPs, as shown in Fig. 12b, are frames containing no crayfish, for which the vision system erroneously identified one and marked it with a green dot. FNs, as shown in Fig. 12c are frames containing a

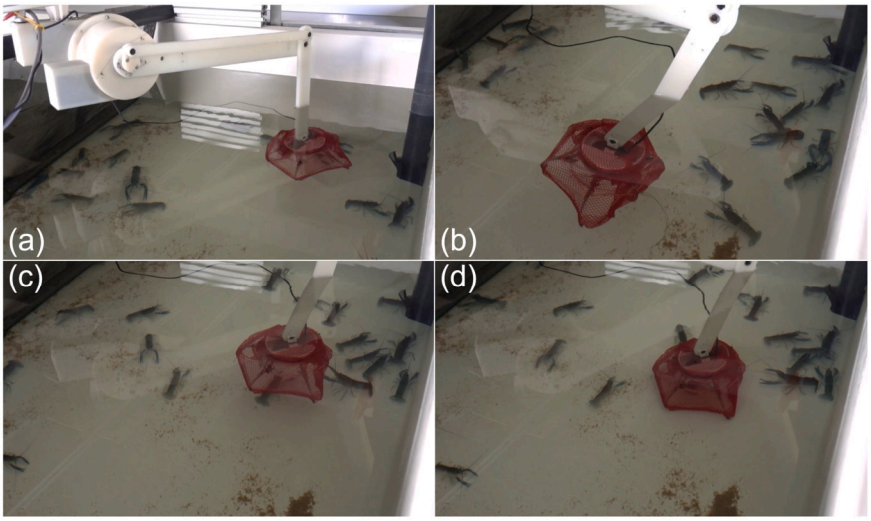
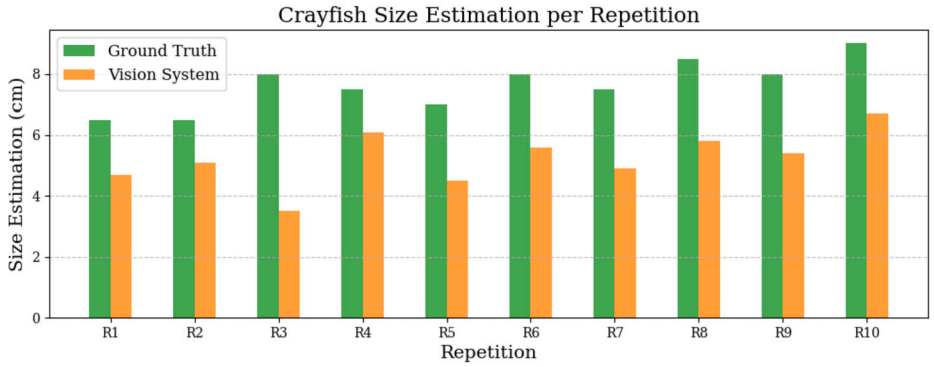


Fig. 10. RoSC system in action during farm experiments (a) System is started (b) The system in scanning state with slow velocity not to scare crayfish. (c) The crayfish detected, the system switches to caging state. (d) Caging state, the manipulator goes down and entraps the crayfish from above.



	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
Ground Truth	6.5	6.5	8	7.5	7	8	7.5	8.5	8	9
Vision System	4.7	5.1	3.5	6.1	4.5	5.6	4.9	5.8	5.4	6.7

Fig. 11. Comparison of the crayfish length estimated by the vision system vs manual measurement (ground truth).

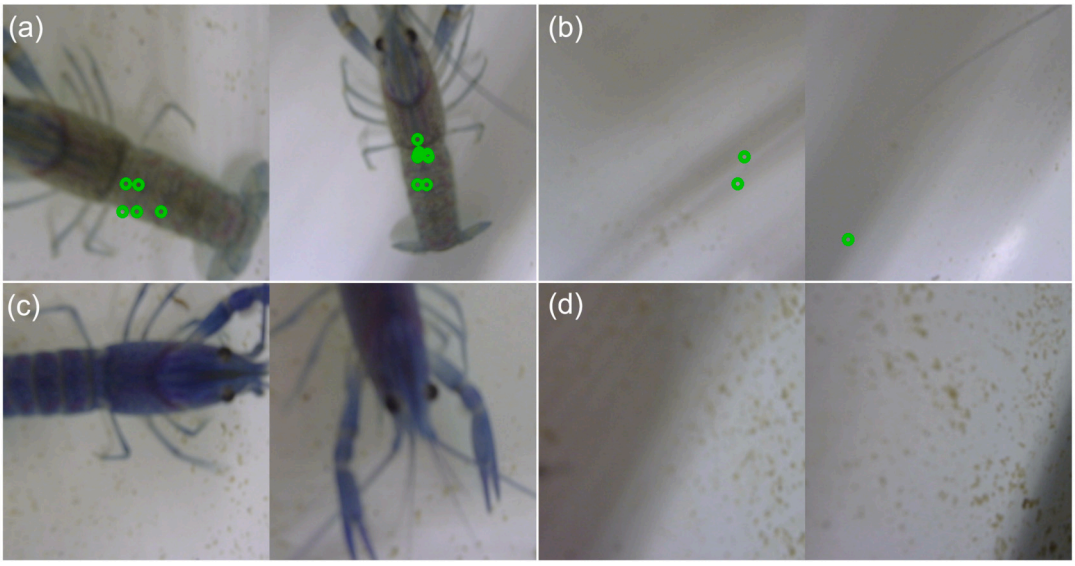


Fig. 12. Examples of onboard camera frames resulting in different detection output. (a) True Positive, the system correctly identifies the crayfish in the frame. (b) False Positive, the system detects a crayfish in an empty frame. (c) False Negative, the system does not detect the crayfish in the frame. (d) True Negative, the system does not detect any crayfish in an empty frame.

part or a full crayfish erroneously not identified by the system. Finally TNs, as shown in 12d, are frames containing no crayfish and for which the vision system correctly does not detect any. Table 7 reports the confusion matrix obtained with the aforementioned definitions. The high number of FNs resulted in the system often not performing the reaching behaviour in presence of a crayfish. Conversely, the 63 FPs resulted in the system attempting a caging in absence of a crayfish. Finally, among the 49 TPs, several were obtained on the same specimen, and overall the system attempted 10 caging attempts on as many crayfish. Among the attempts, 9/10 resulted in the manipulator to get moving to cage the crayfish, with the unsuccessful attempt caused by a poor synchronization between the control and vision system nodes. The number of successful caging events was 2/10 corresponding to 20%. Unsuccessful caging was due to different reasons. In some cases, the crayfish run away before the gripper reached the floor of the tank, whereas in other cases it simply kept moving and the gripper reached the portion of the tank at which the crayfish was actually detected. In general, we observed a time lag of over 1 s between the detection of the crayfish and the activation of the arm, which partially explains the lower detection rate with comparison with test 2.

Two videos showing the successful and failed caging attempts are available in the Supplementary Material.

8. Discussion and conclusion

This paper presented a solution to automatically sort crayfish by size in aquaculture facilities. The design of the system followed a systematic user-driven approach in close collaboration with a crayfish farming company, and it resulted from the trade-off among several requirements (See HoQ Supplementary Materials). The company was interested in a portable system that could be easily mounted on their existing (or on perspective) tanks, work in a humid/wet environment, and manipulate the crayfish without causing them harm or stress. The resulting solution was a waterproof robotic manipulator with a caging gripper end effector with a camera integrated in the palm, and a vision system capable of detecting crayfish and estimate their size which can be mounted directly on the top of the tanks.

Differently from previous studies [23], our structured co-design approach enabled an easy deployment of the system, helped concept design generations, and allowed an effective re-use of existing elements. Indeed, thanks to comparison with state of the art solution, the kinematic design of an underwater legged robot has been used as starting point [30] with substantial redesign of the cannisters to reduce the number of penetrations, decrease the weight and cost of the components, and re-distribute the inertia. Crayfish behavioural studies and interaction with the farmers also identified a top-caging approach as the optimal for trapping, without distressing, the animals. This shaped the design concepts, and we selected a passive wrist with a gripper with an in-palm camera, flexible fingers, and enclosing net for minimizing actuations, penetrations, costs and complexity. Without a structured and effective interaction between the two parts, such combinations would not have emerged.

The requests of the company represent a considerable novelty for the sector of crustacean aquaculture in which, while automation already plays a main role, it is mostly dedicated to monitoring and feeding tasks [2,3,6,7]. Instead, in this work, we developed and tested a system capable of catching and handling live, non-anesthetized crustaceans. Catching fish with a robot is not just a concern of aquaculture industry, but also of marine biology for in situ specimen collection. In a recent essay [8], reflected on the use of robotics in deep sea ichthyology, stating that the technology at the time was not yet capable of catching swimming fishes. Since the essay, the technology has evolved and roboticists realized the importance of a bio-inspired design approach which involved a deeper study of the animals, their habits and locomotion, and the use of innovative materials for construction yielding adaptability and intrinsic safety in manipulation [13]. However, the examples of robotic manipulation of live marine animals reported in literature so far dealt with slow animals such as jellyfish, sea hurcins or sea cucumbers [37].

The close interaction with Noola Redclaw allowed us to observe and study the behaviour of crayfish in their aquaculture environment and perform the preliminary tests described in section 2.3 which lead to the selection of the caging gripper design. The tests 1 and 2, reported in section 6.2, confirmed what had been already observed during the preliminary tests, i.e. a slow manipulator velocity is critical not to scare off the animals and trigger their fast escaping manoeuvre. The hardware and kinematic control of the robotic manipulator and gripper, has been validated both during lab and field trials, with no malfunctioning observed. The vision system has also been validated in both scenarios, and we have observed a tendency to underestimate the length of the crayfish. The main reason for such mismatch is due to the dataset used for the training of the neural network, which used images of real crayfish in water with a scale. Indeed, due to the natural curvature of the crayfish body, the resulting size estimation is around 30% lower than the ground truth measurements taken on the fully stretched bodies. The effect of such underestimation was observed in the lab trials, for which the highest number of False Positives was observed when targeting the larger crayfish, which tended to be confused for smaller sized ones. This results can be improved by retraining the network using the ground truth measurements, or by simply considering a systematic length correction equal to the relative error reported in 6.2. The fully automated operation carried out in test 4 (section 6.2) highlighted a notable difference between the preliminary performance of the vision system and the actual performance in the farm during operations. A high number of FPs was observed, with shadows and algae being mistaken for a crayfish. On the other hand, a relatively high number of FNs was also reported. The majority of FNs are obtained in frames such as those shown in Fig. 12c in only a part of crayfish is present and the system could not identify it. This discrepancy stems from the differences between videos used for training, and the video from the onboard camera during the trials. The videos used for training show a wider field of view with one or more whole crayfish present in the same frame. On the other hand, as depicted in Fig. 12, the on board camera has a much narrower view, crayfish are rarely entirely displayed in a frame and the system is not capable of detecting them. In case of TPs, the system attempted caging on actual crayfish 10 times and we observed a 20% successful caging. Two main causes for such results were identified. On one hand, the time lag between the detection and the activation of the manipulator caused several misses. On the other hand, crayfish exhibited significantly higher mobility with respect to the preliminary studies described in section 2.3, possibly because their rocky shelters were removed from the experimental tanks. Several misses occurred because the crayfish were already moving at the time of detection and when the manipulator reached the position, the animals had already moved past it. This was not observed during tests 1 and 2, because the human operator was implicitly estimating the trajectory of the crayfish and started the reaching control routine at the best moment. The system could be improved by adding the crayfish tracking to the vision system and developing a closed-loop Caging behaviour capable of reacting to the crayfish movements, or to estimate their trajectory to intercept them.

To conclude, the system design presented in this paper successfully demonstrated its functionality to detect and cage live crayfish in aquaculture tanks, while simultaneously meeting the requirements of the end users. The mechanical hardware comprising the arm and gripper, and the low level control, worked flawlessly in a wet environment and never caused damage to the crayfish that were handled. On the other hand, the vision system could be improved by correcting the crayfish size underestimation and implementing the tracking of crayfish to improve the performance on more mobile individuals.

CRediT authorship contribution statement

Giacomo Picardi: Investigation, Methodology, Software, Writing – original draft, Writing – review & editing. **Anna Astolfi:** Investigation, Methodology. **Karthik Seemakurthy:** Methodology. **Bradley Hurst:** Methodology. **Elena Piana:** Conceptualization, Investigation. **Petra Bosilj:** Investigation, Writing – original draft, Methodology. **Marcello Calisti:** Funding acquisition, Conceptualization, Methodology, Writing – review & editing, Investigation.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used ChatGPT in order to improve readability and language of the work. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work has been partially supported by: the Department for Environment, Food & Rural Affairs (Defra), within the context of Robotics for the automatic Sorting of Crustaceans (RoSC) project, under the framework UK Seafood Innovation Fund (FS135); Research England (Lincoln Agri-Robotics) as part of the Expanding Excellence in England (E3) Programme. The view expressed in this publication are of the authors solely, and not necessarily those of the funders.

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.atech.2025.101168.

Data availability

Data will be made available on request.

References

[1] FAO, The State of World Fisheries and Aquaculture 2022. Towards Blue Transformation. Rome, FAO, https://doi.org/10.4060/cc0461en, 2022.
[2] D. Li, C. Liu, Z. Song, G. Wang, Automatic monitoring of relevant behaviors for crustacean production in aquaculture: a review, Animals 11 (9) (2021) 2709.
[3] J. Reis, R. Novriadi, A. Swanepoel, G. Jingping, M. Rhodes, D.A. Davis, Optimizing feed automation: improving timer-feeders and on demand systems in semi-intensive pond culture of shrimp litopenaeus vannamei, Aquaculture 519 (2020) 734759.
[4] A. Drengstig, A. Bergheim, Commercial land-based farming of European lobster (homarus gammarus l.) in recirculating aquaculture system (ras) using a single cage approach, Aquac. Eng. 53 (2013) 14–18.
[5] A. Campbell, G. Harman, D. John, System for harvesting crustaceans, Tech. Rep., US20060124071A1, United States, 2023.
[6] I. Kuklina, A. Kouba, P. Kozák, Real-time monitoring of water quality using fish and crayfish as bio-indicators: a review, Environ. Monit. Assess. 185 (2013) 5043–5053.
[7] C. Le Cocq, E. Paiva, A. Bensetra, J. De Sonneville, K.-J. Van der Kolk, D. Lejon, M.-L. Teisseire, M. Leonard, C. Sweetlove, Utilization of a gender-based sorting machine for crustacean selection in bioconcentration studies with the freshwater amphipod hyalella azteca, Environ. Toxicol. Chem. 42 (5) (2023) 1075–1084.
[8] G.R.G. Jr., You can't catch a fish with a robot, Gulf Carib. Res. 27 (1) (2016) ii–xiv.
[9] S. Sivčev, J. Coleman, E. Omerdić, G. Dooly, D. Toal, Underwater manipulators: a review, Ocean Eng. 163 (2018) 431–450.
[10] G. Picardi, A. Astolfi, D. Chatzievangelou, J. Aguzzi, M. Calisti, Underwater legged robotics: review and perspectives, Bioinspir. Biomim. 18 (3) (2023) 031001.
[11] D. Edwards, Crayfish escape, in: Oxford Research Encyclopedia of Neuroscience, 2017.
[12] E.R. Larson, J.D. Olden, et al., Field sampling techniques for crayfish, Biol. Ecol. Crayfish 287 (2016) 324.
[13] A. Mazzeo, J. Aguzzi, M. Calisti, S. Canese, F. Vecchi, S. Stefanni, M. Controzzi, Marine robotics for deep-sea specimen collection: a systematic review of underwater grippers, Sensors 22 (2) (2022) 648.
[14] X. Yang, S. Zhang, J. Liu, Q. Gao, S. Dong, C. Zhou, Deep learning for smart fish farming: applications, opportunities and challenges, Rev. Aquacult. 13 (1) (2021) 66–90.
[15] Y. Hasan, K. Siregar, Computer vision identification of species, sex, and age of Indonesian marine lobsters, INFOKUM 9 (2, June) (2021) 478–489.
[16] S. Wang, J. Guo, S. Guo, Q. Fu, J. Xu, Study on real-time recognition of underwater live shrimp by the spherical amphibious robot based on deep learning, in: 2022 IEEE International Conference on Mechatronics and Automation (ICMA), IEEE, 2022, pp. 917–922.
[17] S.A. Vo, J. Scanlan, P. Turner, An application of convolutional neural network to lobster grading in the southern rock lobster supply chain, Food Control 113 (2020) 107184.
[18] S.A. Vo, J. Scanlan, P. Turner, R. Ollington, Convolutional neural networks for individual identification in the southern rock lobster supply chain, Food Control 118 (2020) 107419.
[19] X. Ye, Y. Liu, D. Zhang, X. Hu, Z. He, Y. Chen, Rapid and accurate crayfish sorting by size and maturity based on improved yolov5, Appl. Sci. 13 (15) (2023) 8619.
[20] G. Jocher, Ultralytics yolov5, https://doi.org/10.5281/zenodo.3908559, https://github.com/ultralytics/yolov5, 2020.
[21] X. Yang, J. Yan, Arbitrary-oriented object detection with circular smooth label, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16, Springer, 2020, pp. 677–694.
[22] V. Marinoudi, L. Benos, C. Camacho Villa, M. Lampridi, D. Kateris, R. Berruto, S. Pearson, C.G. Sørensen, D. Bochtis, Adapting to the agricultural labor market shaped by robotization, Sustainability 16 (16) (2024) 7061.
[23] C.W. Bac, E.J. Van Henten, J. Hemming, Y. Edan, Harvesting robots for high-value crops: state-of-the-art review and challenges ahead, J. Field Robot. 31 (6) (2014) 888–911.
[24] J.F. Elfferich, D. Dodou, C. Della Santina, Soft robotic grippers for crop handling or harvesting: a review, IEEE Access 10 (2022) 75428–75443.
[25] M.M. Foglia, G. Reina, Agricultural robot for radicchio harvesting, J. Field Robot. 23 (6–7) (2006) 363–377.
[26] E. Van Henten, D. Van't Slot, C. Hol, L. Van Willigenburg, Optimal manipulator design for a cucumber harvesting robot, Comput. Electron. Agric. 65 (2) (2009) 247–257.
[27] H. Hwang, S.-C. Kim, Development of multi-functional tele-operative modular robotic system for greenhouse watermelon, in: Proceedings 2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2003), vol. 2, IEEE, 2003, pp. 1344–1349.
[28] S. Sakai, M. Iida, K. Osuka, M. Umeda, Design and control of a heavy material handling manipulator for agricultural robots, Auton. Robots 25 (2008) 189–204.
[29] G. Picardi, M. De Luca, G. Chimienti, M. Cianchetti, M. Calisti, User-driven design and development of an underwater soft gripper for biological sampling and litter collection, J. Mar. Sci. Eng. 11 (4) (2023) 771.
[30] G. Picardi, M. Chellapurath, S. Iacoponi, S. Stefanni, C. Laschi, M. Calisti, Bioinspired underwater legged robot for seabed exploration with low environmental disturbance, Sci. Robot. 5 (42) (2020) eaaz1012.
[31] B.C. Russell, A. Torralba, K.P. Murphy, W.T. Freeman, Labelme: a database and web-based tool for image annotation, Int. J. Comput. Vis. 77 (2008) 157–173.
[32] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2961–2969.
[33] J. Redmon, A. Farhadi, Yolov3: an incremental improvement, preprint, arXiv:1804.02767, 2018.
[34] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft Coco: Common Objects in Context, Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, Springer, 2014, pp. 740–755.
[35] M. Everingham, L. Van Gool, C.K. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, Int. J. Comput. Vis. 88 (2010) 303–338.
[36] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, preprint, arXiv:1412.6980, 2014.
[37] D.F. Gruber, R.J. Wood, Advances and future outlooks in soft robotics for minimally invasive marine biology, Sci. Robot. 7 (66) (2022) eabm6807.