Journal of Micro and Bio Robotics (2024) 20:3
https://doi.org/10.1007/s12213-024-00167-y

RESEARCH

# Robotic sorting of zebrafish embryos

**Alioune Diouf[1,2] · Ferhat Sadak[1,3] · Edison Gerena[1] · Abdelkrim Mannioui[5] · Daniela Zizioli[6] · Irene Fassi[4] · Mokrane Boudaoud[1] · Giovanni Legnani[2] · Sinan Haliyo[1]**

Received: 2 February 2024 / Revised: 1 April 2024 / Accepted: 15 April 2024 / Published online: 18 May 2024
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2024

**Abstract**
Transcriptomics and metabolomics, two biological research fields that need large numbers of zebrafish embryos, require the removal of unfertilised or nonviable zebrafish embryos. Biologists routinely conduct the tedious, error-prone, and time-consuming manual sorting of embryos. We suggest a novel approach that combines deep learning and microfluidics for automated sorting to overcome this difficulty. To determine the developmental stage and viability of zebrafish eggs, we trained an optimized YOLOv5 model with 95.8% accuracy and a processing speed of 10.6 ms per frame, classifying them as dead, unfertilised, or alive. The eggs are contained in traps on a microfluidic chip using micro-pumps. After that, the deep learning system can identify and automatically sort the eggs according to their viability by positioning this chip on an XYZ motorized stage. The sorting experiment was conducted in two modes: without feedback and with feedback while using the dead egg position. The first one had a sorting success rate of 90% as opposed to 97.9% for the feedback mode with 3 seconds required for each dead egg. This automated approach provides a precise and efficient way to handle a large number of zebrafish embryos while also greatly reducing the workload associated with manual sorting. The success rates attained demonstrate the usefulness and effectiveness of our suggested methodology, opening new avenues for biological research involving accurate embryo selection.

**Keywords** Zebrafish embryo · Deep learning · Micro robotics · Microfluidic · YOLOv5

## 1 Introduction

Research often involves the use of patient cells or tissue samples. However, to determine whether a mutation in a specific gene can cause a patient's symptoms, experimental animal models are often necessary. The zebrafish (Danio rerio) has become an important model organism for developmental genetic studies and drug discovery. Research based on the zebrafish has led to new advances in numerous medical fields. The analysis of Zebrafish mutants is crucial in the study of hematopoietic, cardiovascular, and vascular disorders, as well as tumours, neurodegenerative, and neuromuscular diseases such as Alzheimer's syndrome, Huntington's disease, and Duchenne muscular dystrophy. These mutants are used to simulate human pathologies to study effective pharmacological therapies.

However, handling zebrafish embryos manually is a time-consuming and tedious task due to their small size and the large quantity of samples required in experiments [1]. Therefore, there is a critical need for a fast and automated screening method to improve working conditions with zebrafish embryos and larvae. Retrieving and sorting eggs after a spawning event and removing the unfertilised or dead ones is crucial for maintaining optimal growth conditions and promoting the well-being of growing embryos. Contaminated hatching water can stimulate bacterial proliferation among the eggs, which secrete enzymes to degrade the eggshells, leading to premature hatching and death. Fungi spores can develop and spread over dead eggs, eventually contaminating healthy eggs and compromising embryo development. Therefore, it is crucial to remove unfertilised eggs or dead embryos to prevent batch contamination, as spoiled eggs can serve as a growth medium for deadly microorganisms [2]. Including unfertilised eggs or premature dead embryos in a cohort of tested embryos can lead to impre-

Ferhat Sadak, Edison Gerena, Abdelkrim Mannioui, Daniela Zizioli, Irene Fassi, Mokrane Boudaoud, Giovanni Legnani and Sinan Haliyo contributed equally to this work

✉ Alioune Diouf
diouf@isir.upmc.fr

Extended author information available on the last page of the article

cise results, as unfertilised eggs will increase the death rate following a screening assay. Therefore, it is recommended to perform early sorting of embryos not only to prevent batch contamination but also to exclude any irrelevant samples from the experiment [3]. The current state of technology for automated microscopic imaging of zebrafish and interpreting these images is well-established. However, automating the sample preparation process offers further opportunities for unique developments [4]. To assess the trend in the cell preparation domain, powerful strategies to sort cells utilizing high spatial and often time-resolved data have emerged. These strategies are collectively called Image-Based Cell Sorting (IBCS) [5]. These IBCS platforms address the major limitations of commonly used cell sorting technologies, such as Fluorescence-Activated Cell Sorting (FACS) and Magnetic Activated Cell Sorting (MACS). These are the most commercialized methods for sorting cells. With the advent of the Big Data Era and the development of Deep Learning (DL), image detection is becoming the mainstream for single cell analysis. DL techniques improve the detection procedure compared to conventional techniques by reducing computational time and detecting irregular shapes, which are common among biological cells [6].

The researchers have developed a classification software that Support Vector Machines (SVM) able to differentiate between wild-type embryos and mutant individuals with an accuracy of 79-99% depending on the test and system [7]. Furthermore, a sorting system based on Convolutional Neural Networks (CNN) is capable of sorting and placing individual zebrafish eggs in multiwell plates [8]. A fully automated pipetting sorting system based on template-matching algorithms made it possible to classify zebrafish eggs and sort them with a robot [9]. Several machine learning approaches are available for reliable detection results but Deep Neural Networks (DNN) is the most popular system exploited so far for zebrafish egg detection [6, 10, 11]. EmbryoNet is a software based on a DNN model to identify zebrafish embryo signaling mutants in an unbiased manner [12]. A system based on Inception v3 is capable of detecting and performing microinjection on zebrafish eggs [13]. A machine vision-guided robot based on YOLOv4 is developed for fully automated embryonic detection and microinjection [14].

Microfluidics-based approaches are still in their early stages but are of significant importance. Once the design of microfluidic chips is established, their fabrication becomes economically viable, potentially reducing the price barrier for widespread adoption. Lab-On-Chip (LOC) represents a new direction in investigative tools that may miniaturize and revolutionize research in toxicity and physiology in vivo [15].

Microfluidics, with its ability to accommodate a variety of actuation techniques including flow control based approaches, is an interesting non-contact method for manipulating cells [16]. However, there is currently no guarantee that the field will not damage or adversely affect the target cell when using external actuation such as electrical or optical methods. It ensures that the cells remain in the liquid medium throughout the process by using only the force of the fluid on the cells. It is possible to create chambers for cell immobilisation with the ability to design microfluidic chips. The combination of microfluidic approaches with DL has the potential to automate the sorting and selection of zebrafish embryos, replacing the manual sample manipulation currently performed by researchers [4]. Microfluidic chip has been used with zebrafish embryos but more for culture, perfusion systems [17, 18].

Although zebrafish egg classification systems are well-established in the literature, there has been little attention and exploration given to fast and precise zebrafish egg sorting systems.. In this paper, we propose fully automated microfluidic and DL based robotic sorting of zebrafish embryos.

## 2 Materials and methods

### 2.1 System overview

A DL model for zebrafish allows for the sorting of dead or unfertilised embryos and viable embryos of stage 1 or other stages of development. The process consists in two phases: the phase of filling the traps of the microfluidic chip with zebrafish embryos followed by the robotic sorting phase.

#### 2.1.1 Filling process

In this phase, zebrafish embryos, measuring approximately 1 mm, are carefully positioned within cavities to streamline and prepare for subsequent sorting procedures. The eggs are propelled through the water within the microfluidic chip's channel by pumps (pump 1 and pump 2) and subsequently captured in purpose-built traps. (see Fig. 1). *Bartels mikrotechnik's* piezoelectric mp6-liq micropumps were used to transport the embryos into the chip. The flowrate of the pumps is controlled by a microcontroller and a driver provided by the pumps manufacturer. By acting on the voltage and the frequency of the driver we can modify the flowrate as shown in Fig. 2.

A single egg can fill a trap, and the others can slide over it to pass through the other traps (see Fig. 3). This method has been shown to be effective in many previous works [19]. The immobilization of zebrafish embryos in microfluidic devices relies on the forces generated by fluid flow within the microchannel. The trapping cavity holds the embryos in

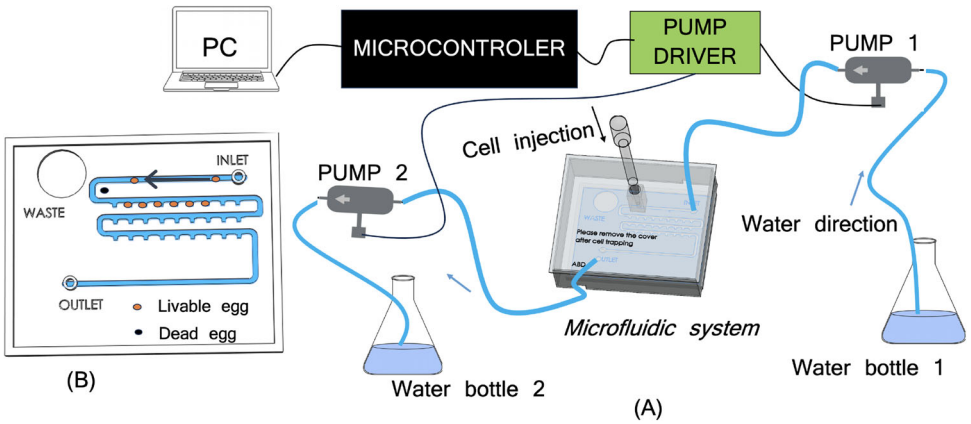


**Fig. 1** PHASE 1 : (A) Overview of the filling system, egg are injected with the pipette from the top , (B) Close look-up to the chip when eggs are injected. Reprinted with permission from IEEE, Copyright (2024), 5719300756547 [21]

place. The transport passageway enables fluid flow, which carries a force component directed towards the trapping cavity, effectively trapping the embryo. The magnitude of this component force is closely related to the geometric parameters of the trapping cavity. Figure 4(B) shows the geometric parameters and their association with the size of the trapping component. These parameters may include dimensions such as width, length, and depth of the trapping cavity, among others. Tang et al. [20] describe that the ability to trap embryos depends on the flow resistance along two pathways:

- Path 1 ($Rx$): Flow resistance along the main channel.
- Path 2 ($Ry$): Flow resistance along the trapping cavity direction

According to [19], the flow resistance along the main channel (Path 1) and along the trapping cavity direction (Path 2), can be described as:

$$\frac{R_x}{R_y} = \left[ \frac{w_c^4}{H_c L_t \left(W_c + H_c\right)^2} + \frac{w_c^4}{H_g L_g \left(W_c + H_g\right)^2} \right] \cdot \frac{H_c L_m \left(W_m + H_c\right)^2}{W_m^4} \tag{1}$$

where $W_{(.)}$ is the width of the channel, $H_{(.)}$ is height, and the $L_{(.)}$ is the length. Letter $m$ , $c$ , and $t$ stand for the main channel, trapping cavities, and transport passageway (behand the cavities) separately. Zebrafish embryos can be successfully trapped in microfluidic systems when the flow resistance along both the main channel and the trapping cavity direction is appropriately balanced. This balance ensures that the fluid flow exerts enough force to direct the embryos towards and eventually trap them within the designated cavity. The geometric parameters of the trapping cavity can be adjusted to influence this balance and improve the trapping efficiency.

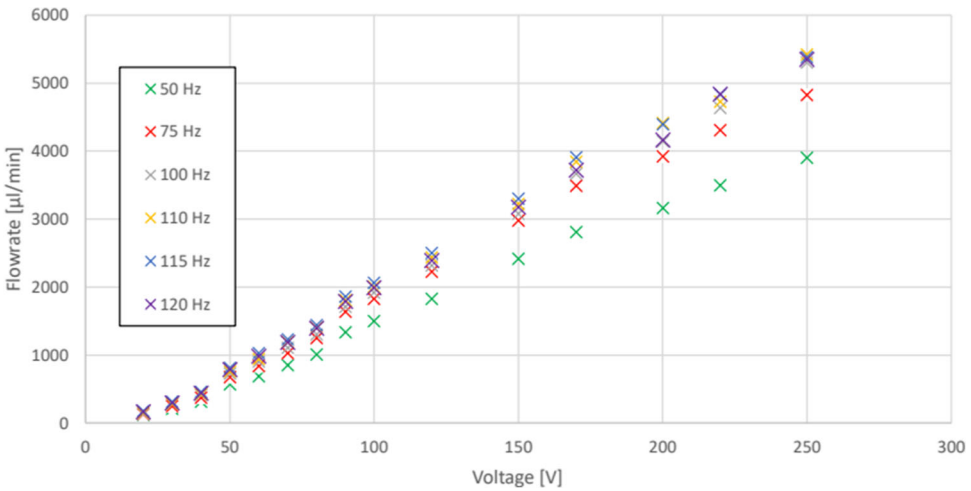


**Fig. 2** mp6-liq micropump flowrate as a function of frequency and voltage. (https://www.bartels-mikrotechnik.de/en/download-area-en/)

**Fig. 3** Zebrafish embryos positioned in the cavities of the chip, one of which is moving to reach a free cavity

To ensure that the trapping cavity can provide enough force to the embryo during the trapping process, it's necessary to adjust the cavity-related parameters ($H_g$, $W_c$, $L_g$, $L_t$) in such a way that the flow resistance ratio $Rx : Ry$ is greater than 1. To ensure effective trapping of embryos, the flow resistance along the trapping cavity ($Ry$) direction must be lower than the flow resistance along the main channel ($Rx$). The channel parameters should be adjusted according to Eq. 1 to achieve the desired trapping and release behaviors of the embryos. The specific values for these parameters found for this study are in Table 1.

**Table 1** The parameters of the channel

| Parameters | Size/mm |
|---|---|
| Lm | 20 |
| Wm | 3 |
| Lg | 1 |
| Lt | 1.5 |
| Hc | 1.5 |
| Wc | 1.5 |
| Hg | 0.5 |

### 2.1.2 Robotic sorting process

The cover is carefully removed and the chip is placed on a motorised XYZ stage after the eggs have been trapped on the microfluidic chip (see Fig. 5). The stage is positioned above a DL system, which moves the microfluidic chip so that each trap can be examined. The zebrafish egg classification algorithm is then used to determine whether to remove or keep the egg in question. A micromanipulator with a glass pipette is used as an effector to suction and hold the egg before finally placing it in the waste section of the chip (see Fig. 6). The process of automated suction involves using a piezoelectric water pump connected to the holding pipette. The pump is activated when the holding pipette reaches the designated pick-up point. The sorting steps are detailed in the algorithm shown in Fig. 15. The DL model training was conducted using *Google Colab's Tesla T4 GPU*. Figure 7 shows the different devices used for the sorting process. The experiment was carried out with the uMp manipulator from *Sensapex*, the XYZ-stage LNR50D and DRV250 stepper motors from *Thorlabs*. The vision system comprises a UI-3590CP-C-HQ R2 USB camera from *IDS Imaging Development Systems*, a LM50HC-VIS-SW lens from *Kowa* and OZB-A4515 6W LED lights from *Kern Optics*. *Bartels mikrotechnik's* piezoelectrip mp6-liq micropumps were used for cell suction as well as for the zebrafish embryo filling system in the traps of the microfluidic chip.

## 2.2 Chip design and fabrication

The microfluidic chip consists of an inlet and an outlet, as well as traps designated for cell placement. A waste section

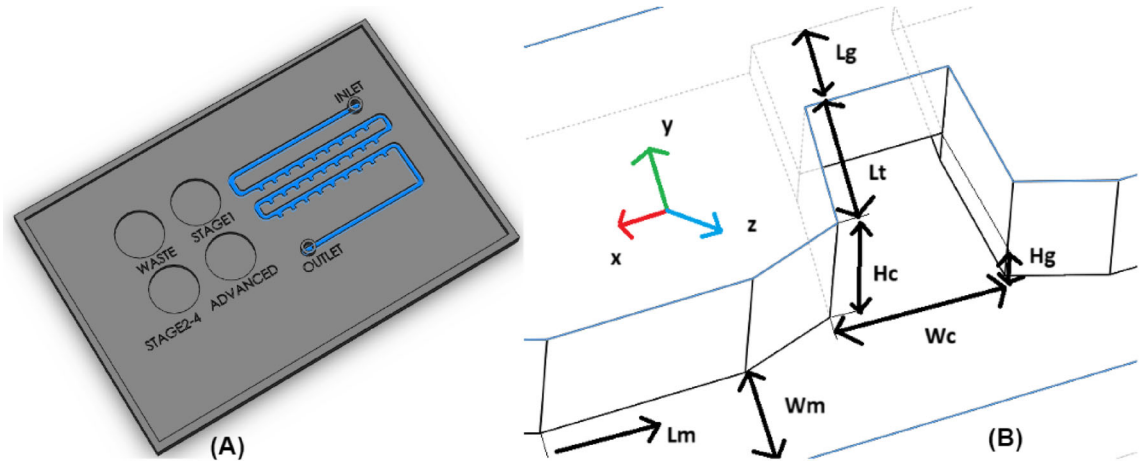


**Fig. 4** (A) Microfluidic chip design. (B) The geometric parameters of a single cavity

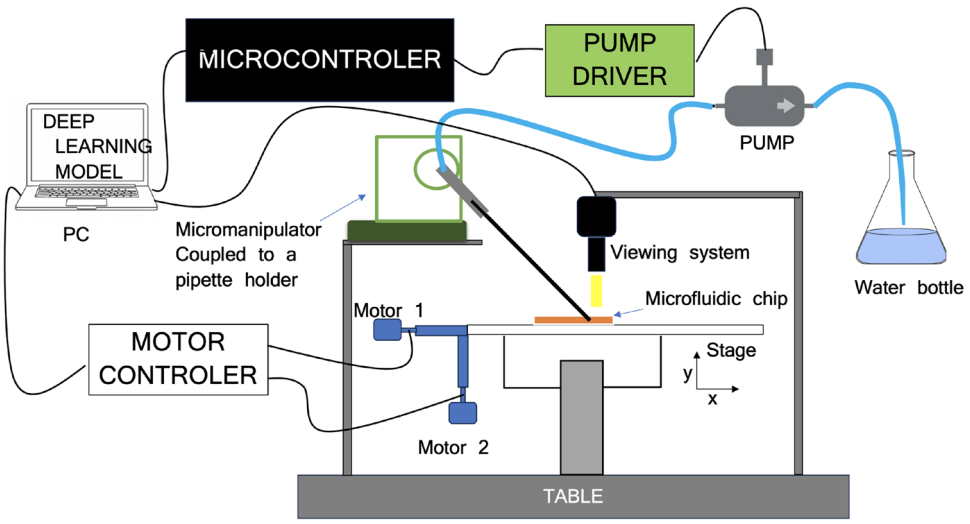


**Fig. 5** PHASE 2 : Schematic overview of the sorting system. Reprinted with permission from IEEE, Copyright (2024), 5719300756547 [21]

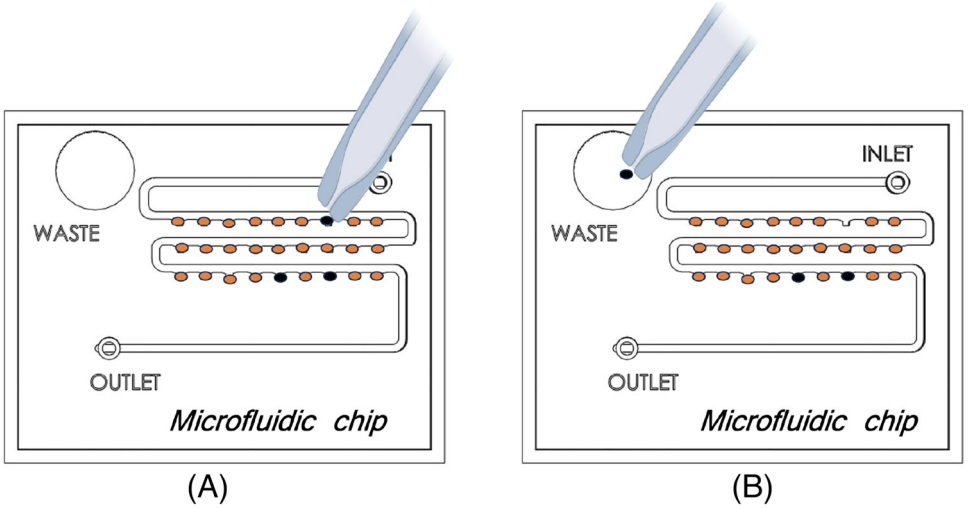


**Fig. 6** Pick and place process: (A) Dead or unfertilised embryo being picked. (B) : placed on waste part. Dead cell are black dots and orange are for livable. Reprinted with permission from IEEE, Copyright (2024), 5719300756547 [21]

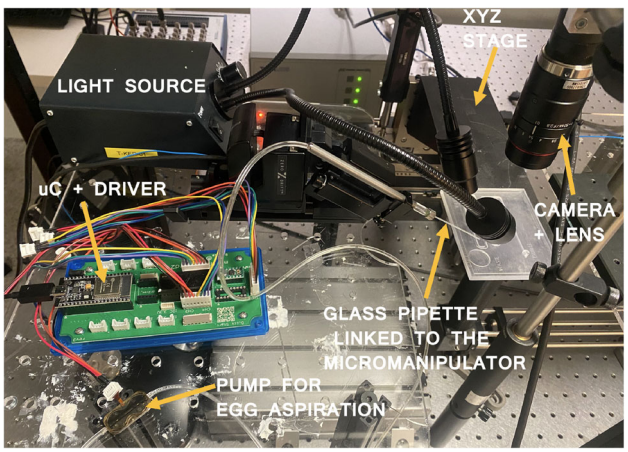


**Fig. 7** Overview of the real system for the experiments. Reprinted with permission from IEEE, Copyright (2024), 5719300756547 [21]

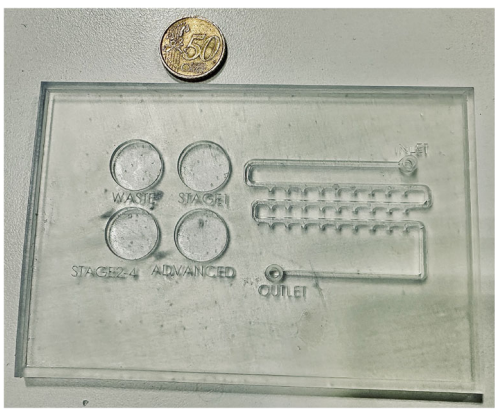

**Fig. 8** Microfluidic chip printed in 3D after cleaning and passing under UV light

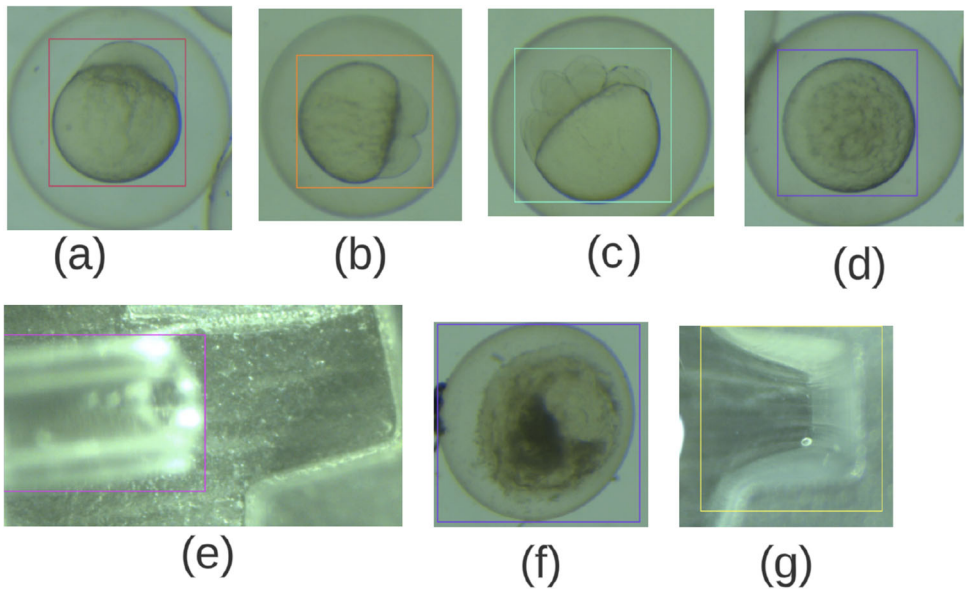


**Fig. 9** Differents classes of the model. $(a) : stage1$ , $(b) : stage2-4$, $(c) : advanced$, $(d) : returned$, $(e) : holder$, $(f) : dead$ . $(g) : empty$. Reprinted with permission from IEEE, Copyright (2024), 5719300756547 [21]

is where non-viable cells are deposited by the micromanipulator. (see Fig. 4). The other parts are filled by cells at the corresponding stage of development. The chip is designed with the 3D CAD software : *Solidworks 2021* and printed using *Formlab's Form 3+* printer with clear V4 resin. After printing, the chip is cleaned with isopropanol and then placed in UV light for 30 min at 60°C (see Fig. 8). A cover, which allows for the injection of eggs on the chip is made with the Fig. 9 same resin and is placed above the chip during the filling process.

## 2.3 Data collection, labelling and augmentation

Wild-type adult zebrafish are maintained acccording to standard protocol in the Aquatic Facility of Sorbonne University [22]. Zebrafish eggs are obtained through natural spawning on the day of each experiment. Zebrafish are known for their prolific egg production, and eggs are typically collected shortly after spawning. The eggs are incubated after collection in E3 solution. It refers to a commonly used growth medium specifically designed for maintaining zebrafish embryos at the developmental stage around 3 days post-fertilization. This medium ($34.8\,g\,NaCl$, $1.6\,g\,KCl$, $5.8\,g\,CaCl_2.H_2O$, $9.78\,g\,MgCl_2.6H_2O$) is crucial for providing the necessary nutrients, ions, and environmental conditions to support embryonic development and viability.

The dataset collection was conducted during the first 5 hours after the immediate collection of fresh zebrafish embryos. The zebrafish embryo images were captured at different developmental stages. Images were collected using a combination of an external USB camera UI-3590CP-C-HQ R2 from the manufacturer *IDS Imaging Development Systems* connected to the microscope *Nikon* SMZ800N. Accordingly, 1381 images were acquired, with a total of 1662 annotations shown in Table 2. For the purpose of dataset preparations, the online tool called *RoboFlow* was used. Images are labelled into seven classes : $'stage1'$ , $'stage2-4'$, $'advanced'$, $'dead'$, $'returned'$, $'empty'$ , $'holder'$.

The classes $'empty'$ and $'dead'$ are under-represented compared to the others. However, this is not an issue because these classes contain images that are invariant in shape, requiring less data for training. As zebrafish studies are often conducted at the primary stage, such as microinjection, it is crucial to detect and differentiate between stages 1, 2, and 4. After 1 hour post-fertilization (hpf), an egg remaining at stage 1 indicates that it is unfertilised. Advanced stages, i.e., above stage 4, are grouped into a single class. The $'empty'$ class represents an empty cavity, and the $'holder'$ class represents the tip of the pipette that will extract the embryos. Augmented data is created by making minor modifications to the origi-

**Table 2** Class balance on dataset

| Classes | Train | Validation | Test | Total |
|---|---|---|---|---|
| stage 1 | 293 | 84 | 48 | 425 |
| stage 2-4 | 265 | 64 | 43 | 372 |
| dead | 81 | 28 | 13 | 122 |
| advanced | 209 | 68 | 31 | 308 |
| returned | 174 | 47 | 32 | 253 |
| empty | 72 | 18 | 12 | 102 |
| holder | 72 | 14 | 9 | 80 |
| all | 1151 | 323 | 188 | 1662 |

nal data. In the case of image augmentation, geometric and color space transformations such as flipping, resizing, cropping, adjusting brightness and contrast are applied to increase the size and diversity of the training set. Rotation, brightness, and saturation operations are performed for our case. Image augmentation techniques, such as random rotation, are commonly used in computer vision tasks to diversify and increase the training dataset, reducing overfitting. Random rotation introduces variability into the training data, enabling the model to learn to recognize objects or patterns from different perspectives, making it more resilient to variations in real-world scenarios where the camera angle may vary. The zebrafish eggs in the water move and rotate randomly due to the flow. Therefore, to enhance the model's performance, it is appropriate to apply rotation augmentation. Additionally, it is recommended to apply brightness augmentation as the model may be used in scenarios with varying brightness levels, such as different microscope images. During training, adjusting the saturation of images can simulate different colour environments and conditions, enabling the model to learn features that are invariant to changes in colour saturation. This can enhance the model's capacity to generalize to real-world situations where colours may differ significantly.

These operations resulted in a more larger, augmented Table 3 training dataset with a total of 2807 images.

## 2.4 Description and evaluation of YOLOv5 algorithm

A recent version of the YOLO (*You Only Look Once*) algorithm: YOLOv5 [23] was employed. YOLOv5 builds upon the previous versions, incorporating various improvements in terms of speed, accuracy, and efficiency.Its architecture is composed :

- Backbone: YOLOv5 uses a CSPDarknet53 backbone as its feature extractor. CSPDarknet53 is a variant of Darknet, which is a deep neural network architecture designed for object detection tasks. CSPDarknet53 includes a "cross-stage partial" (CSP) connection scheme, which enhances feature reuse and gradient flow, leading to improved performance.
- Neck: A novel neck architecture called PANet (Path Aggregation Network) is introduced. PANet is designed to aggregate features at different scales and resolutions efficiently, enabling the model to detect objects of varying sizes effectively.
- Head: The detection head of YOLOv5 consists of several convolutional layers responsible for predicting bounding boxes and object classes. YOLOv5 predicts bounding boxes using anchor boxes and employs a variant of the focal loss function to handle class imbalance during training.

**Table 3** Data augmentation parameters

| Operation | Values |
|---|---|
| Rotation | -15° and 15° |
| Brightness | -25% and 25% |
| Saturation | -25% and 25% |

Overall, YOLOv5 is designed to strike a balance between speed and accuracy, making it suitable for a range of real-time object detection applications, such as autonomous vehicles, surveillance systems, and robotics. Its modular architecture and various variants provide flexibility to meet different computational constraints and application requirements. The speed of YOLOv5 is a crucial factor in the zebrafish embryo sorting system, particularly during the initial stages of development when biological studies such as microinjection need to be conducted before the embryos progress to later stages. The YOLOv5 training process optimizes multiple loss functions to ensure precise object detection. The loss function is the combination of loss functions for the bounding box, classification, and confidence. Equation 2 represents the overall loss function of YOLOv5 :

$$loss_{YOLOv5} = loss_{bbox} + loss_{class} + loss_{conf} \tag{2}$$

Where *conf* refers to confidence, *bbox* to bounding box and *class* to classification. The box loss measures how well the predicted bounding box covers the object and accurately locates its center. It evaluates the spatial accuracy of the predicted bounding boxes. Objectness measures the likelihood that an object is present within a specific region of interest. It helps in distinguishing between true objects and background clutter. The classification loss evaluates how accurately the model predicts the correct class label for the detected object. It assesses the algorithm's ability to classify objects into predefined categories. Common evaluation metrics for DL are precision (P) (Eq. 3), which is precision rate, recall rate (R) (Eq. 4) and mean average precision (mAP) (Eq. 6) and Intersection over Union (IoU) (Eq. 5) . The expressions are as follows:

$$P = \frac{TP}{(TP + FP)} \tag{3}$$

$$R = \frac{TP}{(TP + FN)} \tag{4}$$

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \tag{5}$$

IoU is a measure of how well the predicted bounding box overlaps with the ground-truth bounding box. Among them,

true positives (TP), false positives (FP), and false negatives (FN), represent positive samples with correct classification, negative samples with incorrect classification, and positive samples with incorrect classification, respectively. AP is the average accuracy rate, which is the integral of the P index to the R index; mAP is the average accuracy of the mean, which means that the AP value of each category is summed, and then divided by all categories, i.e., the average value. It is defined as follows:

$$mAP = \frac{1}{|Q_R|} \sum_{q=Q_R} AP(q) \tag{6}$$

A high *mAP* means that the model has both a low false negative and a low false positive rate. The higher the *mAP*, the more precise and the higher the recall is for the model. Additionally, $mAP@0.5$ and $mAP@0.5:0.95$, which assess the mAP over different IoU thresholds from 0.5 to 0.95.

### 2.5 Genetic algorithm for hyperparameter optimization of YOLOv5

Hyperparameter tuning in object detection involves selecting the best values for parameters that shape the model's architecture and training process. These parameters significantly affect the model's accuracy, precision, and recall. The process of hyperparameter tuning entails systematically testing different combinations of these parameters to identify the configuration that maximizes the chosen performance metric, like accuracy or $mAP$. Common techniques for hyperparameter tuning include grid search, random search, Bayesian optimization, and genetic algorithms. Each method aims to efficiently explore the hyperparameter space to find the optimal configuration for the object detection model. Selecting the optimal hyperparameters for YOLOv5 is particularly challenging due to the vast parameter space involved. Initially, we utilized the hyperparameters provided by the official YOLOv5 model as a starting point and fine-tuned them for our custom dataset. Each model underwent training and evaluation using an objective function, where we employed mAP metrics with a specific weighting scheme to ensure consistency with our evaluation method. The basic loss function of YOLOv5 has not been altered. Instead, a genetic algorithm was applied to optimize the YOLO hyperparameters after using the classic YOLO loss function. Models were selected based on a fitness score derived from the evaluation metrics, aiming to maximize this score. The goal is to maximise fitness (see Eq. 7). YOLOv5 has a default fitness function which is a weighted combination of metrics $mAP@0.5$ contributes 10% of the weight and $mAP@0.5:0.95$ contributes the remaining 90%. We used the default fitness function, which is defined as follows:

$$fitness = w_1.P + w_2.R + w_3.mAP@0.5 + w_4.mAP@0.5:0.95 \tag{7}$$

where: $P$ represents Precision, $R$ represents Recall, $mAP@0.5$ represents mean Average Precision at 0.5, $mAP@0.5:0.95$ represents mean Average Precision from 0.5 to 0.95, $w_1, w_2, w_3, w_4$ represent the corresponding weights defined as [0.0, 0.0, 0.1, 0.9] . These models were then subjected to a genetic algorithm mutation operator, which introduced random perturbations to explore new configurations while preserving successful solutions. The iterations continued for 10 generations. The final hyperparameters were chosen based on the YOLOv5 model with the highest evaluation metric score at the end of the iterations. We successfully tuned all 29 hyperparameters for YOLOv5, demonstrating its effectiveness in improving overall detection accuracy. Figure 10 illustrates the hyperparameter tuning process using a Genetic Algorithm, with the fitness score plotted on the y-axis and hyperparameter values on the x-axis, where greater concentrations are highlighted in yellow.

## 3 Results and discussion

The training was completed in 100 epochs, a batch size of 16 using YOLOv5x which is the extra large version of YOLOv5 [23]. This version comes with high robustness at the cost of higher computation time. The pixel size of the input image was set to be $640 \times 640$.

### 3.1 Zebrafish embryo detection results

The training results are summarized in the Table 4. The values of Precision, Recall and mAP are obtained using the equations respectively Eqs. 3, 4, 6. The confusion matrix (see Fig. 11) aids in comprehending the classes that are being misclassified. The $'empty'$ and $'holder'$ classes are the most precise and are not mistaken for any other class due to their dissimilarity. Similarly, the $'dead'$ class is not mistaken because eggs become degraded with whitish or black colors depending on the luminosity applied, which explains why they are not confused. The training loss is a measure used to evaluate how well a deep learning model aligns with the training data. It reflects the model's error on the training set and indicates how well the model is fitting the dataset. It is expected to decrease over time as the model learns from the

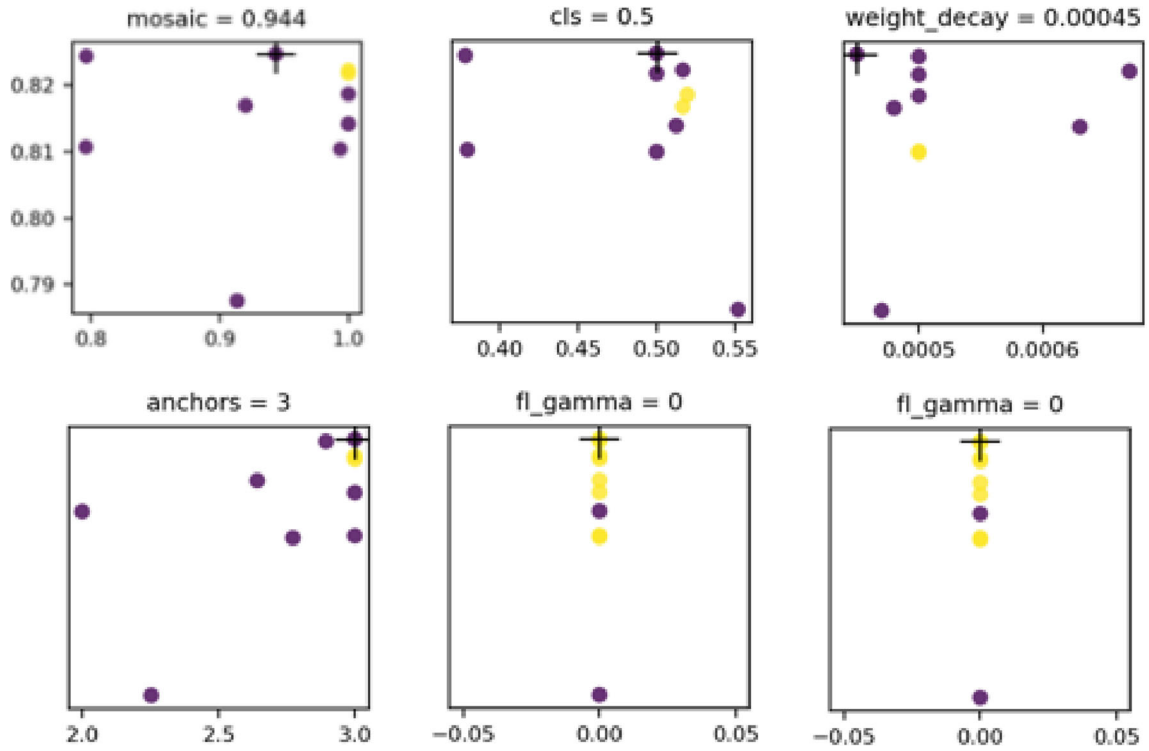


**Fig. 10** Some hyperparameters tuning.Fitness score plotted on the y-axis and hyperparameter values on the x-axis, where greater concentrations are highlighted in yellow

dataset, which is the case in our study (see Fig. 12). However, a very low training loss does not necessarily guarantee good performance on new, unseen data, as the model may have overfit the training data. Therefore, the validation loss is useful in assessing the model's ability to generalize to unseen data during training. Although the two curves converge, there is still a shift, indicating that the model performs well on training data but not on validation data. The initial training resulted in an accuracy of 0.93, but to improve results, we attempt to optimize the hyperparameters.

The optimization of the YOLOv5 model offers several advantages. As shown in Fig. 13 and summarized in the Table 5, the model consistently achieves higher accuracy over generations. Therefore, the training and validation loss for the optimized YOLOv5 demonstrates a significantly better fit for the training and validation datasets. Unlike the default model, this one shows a loss validation curve which follows the loss training towards the end of the epoch. The optimized model provides accuracy of approximately 82.9% for $mAP@0.5:0.95$ and 95.8% for $mAP@0.5$, while the default model provides accuracy of approximately 78.4% for $mAP@0.5:0.95$ and 93% for $mAP@0.5$ after 100 epochs. Which is an increase of 4.5% and 2.8% respectively.

**Table 4** Training results for each class

| Classes | Precision | Recall | $mAP@0.5$ | $mAP@0.5:0.95$ |
|---|---|---|---|---|
| stage 1 | 0.732 | 0.814 | 0.851 | 0.732 |
| stage 2 | 0.717 | 0.844 | 0.866 | 0.767 |
| dead | 0.894 | 0.905 | 0.95 | 0.772 |
| advanced | 0.902 | 0.838 | 0.941 | 0.801 |
| returned | 0.892 | 0.875 | 0.92 | 0.819 |
| empty | 0.865 | 0.83 | 0.921 | 0.798 |
| holder | 0.983 | 1 | 0.995 | 0.853 |
| all | 0.868 | 0.882 | 0.93 | 0.784 |

The optimization of the YOLOv5 model provides several benefits. As illustrated in Fig. 13 and summarised in Table 5, the model consistently achieves higher accuracy across generations. Consequently, the training and validation loss for the optimized YOLOv5 demonstrates a significantly better fit for the training and validation datasets. Unlike the default model, this one exhibits a validation loss curve that follows the training loss towards the end of the epoch. The optimized model achieves an accuracy of approximately 82.9% for $mAP@0.5:0.95$ and 95.8% for $mAP@0.5$, while the default model achieves an accuracy of approximately 78.4% for $mAP@0.5:0.95$ and 93% for $mAP@0.5$ after 100 epochs. This represents an increase of 4.5% and 2.8% respectively. The model was tested on a testing database of 241 images. With the same GPU, an accuracy of 95.8% was achieved with a detection speed of 10.6 ms per frame.

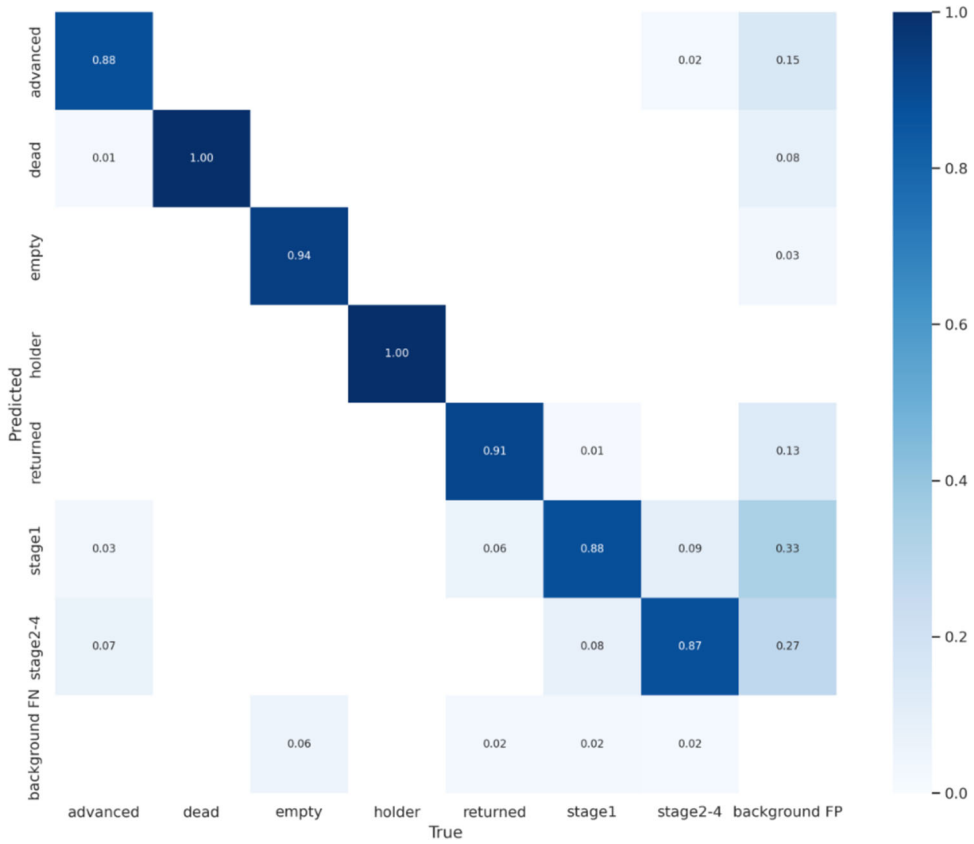


**Fig. 11** Confusion matrix was made at IoU threshold of 0.45, confidence threshold of 0.25

Figure 14 shows examples of zebrafish eggs detected using the trained model. The model can detect various stages of zebrafish egg development, as well as mortality, with confidences above 90%.

## 3.2 Zebrafish embryo sorting experiment result

The optimized model is utilized for robotic sorting zebrafish eggs. The zebrafish eggs utilized in the experiments were obtained under the same conditions as when the image dataset

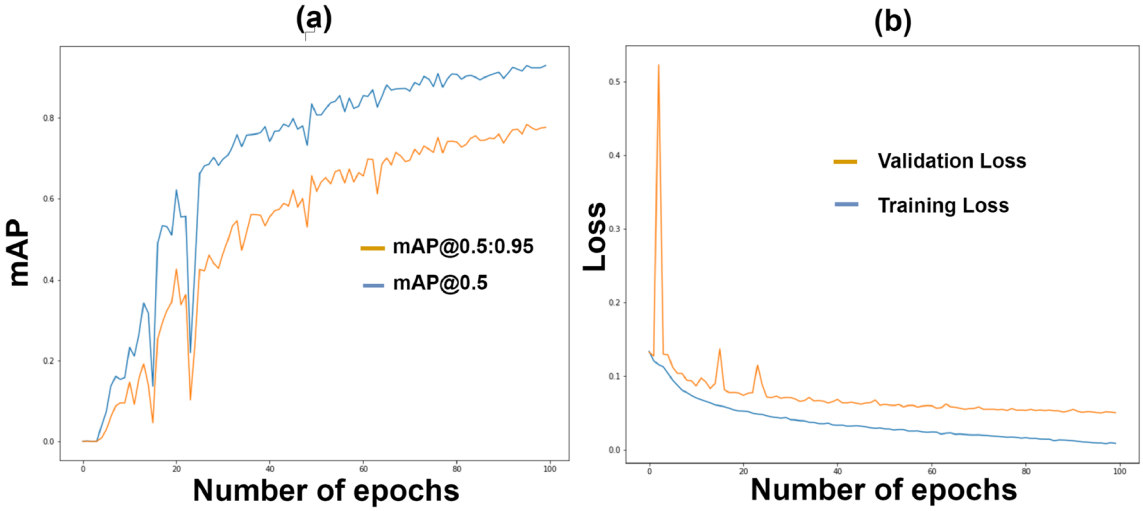


**Fig. 12** Performance of first training. (a) $mAP@0.5$ and $mAP@0.5:0.95$ evolution through generations. (b) overall training loss and validation loss plotted using Eq. 2

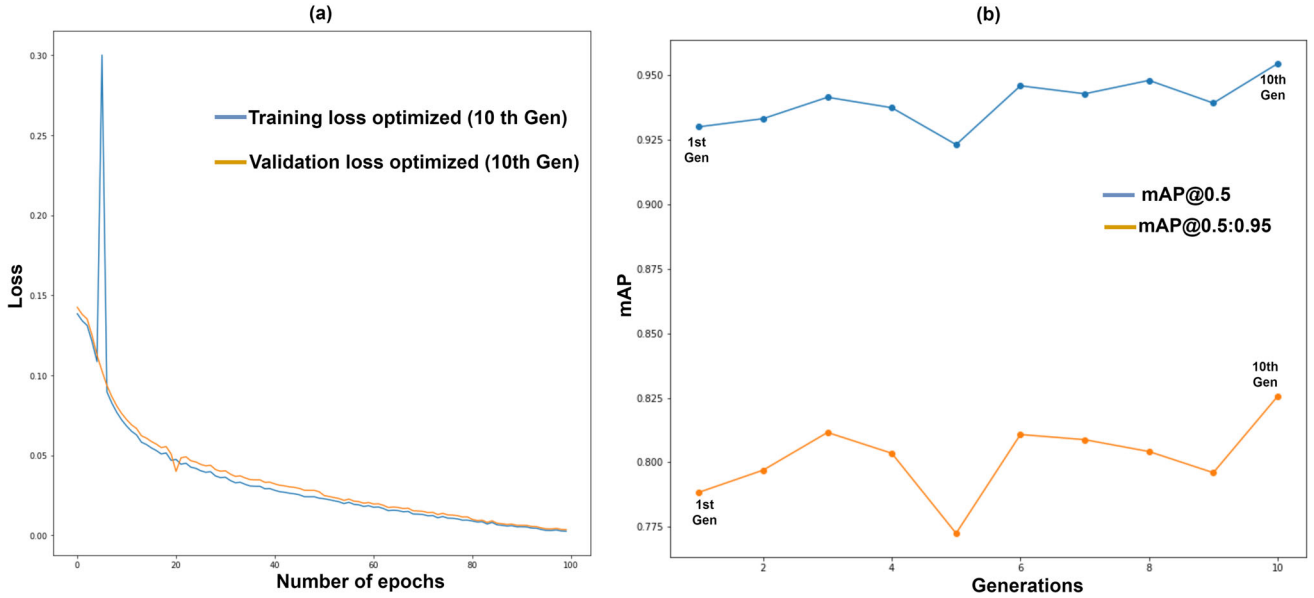


**Fig. 13** (a) Overall training loss and validation loss optmized (10th generation). (b) $mAP@0.5$ and $mAP@0.5:0.95$ evolution through generations

**Table 5** Optimized model results across generations

| Generations | Precision | Recall | $mAP@0.5$ | $mAP@0.5:0.95$ |
|---|---|---|---|---|
| 1 | 0.86711 | 0.9253 | 0.93 | 0.7884 |
| 2 | 0.8394 | 0.93655 | 0.93317 | 0.797 |
| 3 | 0.8671 | 0.8988 | 0.94145 | 0.8116 |
| 4 | 0.85584 | 0.9026 | 0.9374 | 0.80353 |
| 5 | 0.8425 | 0.9088 | 0.9231 | 0.7725 |
| 6 | 0.8619 | 0.9273 | 0.9459 | 0.8108 |
| 7 | 0.85676 | 0.93631 | 0.9428 | 0.8088 |
| 8 | 0.8668 | 0.9512 | 0.9480 | 0.8042 |
| 9 | 0.8858 | 0.8834 | 0.9392 | 0.7960 |
| 10 | 0.9015 | 0.9102 | 0.958 | 0.829 |

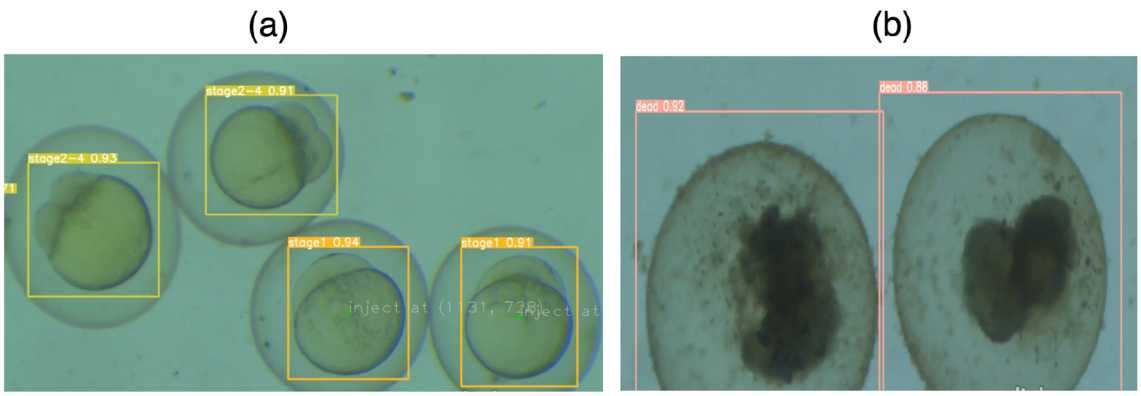


**Fig. 14** (a) Image captured on a zebrafish egg detection video with our optimized model(all egg are livable). (b) Image captured on a zebrafish egg detection video with the optimized model (two dead eggs detected). It is clear that the eggs are well detected as well as their stage of development with great confidence

was created. Dead eggs were separated into a separate tube and used during the experiments. Each trial was conducted with 40 eggs, consisting of 20 dead eggs and 20 viable eggs at different stages of development. At 1 hpf, the eggs were injected into the microfluidic chip using a pipette. Pumps 1 and 2 were set at a flowrate of 1000 $\mu l/min$ to move the eggs, causing them to fall into the holes (see Fig. 3).

#### 3.2.1 Without feedback use of dead egg position

In this mode the XYZ-stage speed is set at 20mm/s and the manipulator at 4mm/s. The microfluidic chip is designed so that the traps all have the same horizontal distance between them called Ds as shown in Fig. 15(c). The XYZ-stage is controlled in step by step mode, so that each step of the XYZ-stage corresponds to a displacement from trap to trap and a focus of the camera on the following trap. During the $'Initialization'$ step of the algorithm (see Fig. 15(a)), the micromanipulator is initialized by fixing Z and Y coordinates so that the holdertip glass faces the initial trap and the Deep Learning model is initialized to detect embryos. We have measured the necessary displacement along X axis (of micromanipulator frame) from the initial position to reach the egg in the trap. This predefined distance is integrated into the coding program for access to an embryo in case of sorting. The first trial showed a major efficiency (100%) in the detection of eggs because all dead eggs and livable eggs were well detected, however, the pick and place procedure had a very low efficiency since only 5 of them was well picked. A significant amount of water is removed from the chip during the trap filling process to prevent displacement of zebrafish embryos after immobilization. To address this issue, the pump was set to bidirectional mode, which enables the injection of the required amount of water for suction and automatic suction of the egg with the injected water in the trap. A second trial was conducted using a pump set to bidirectional mode for the suction process. The pick and place success rate significantly improved from 20% in the first trial to 90%.

After immobilising the eggs, the cover of the chip is carefully removed. The chip is then placed on the XYZ stage, and the sorting process begins. Six trials were conducted. The first two were conducted without feedback using the dead egg position, and the remaining four were conducted with feedback using the dead egg position.

#### 3.2.2 With feedback use of dead egg position

Our experiment for dead egg positioning without feedback provided us satisfactory results. Nevertheless, the micro-displacements of the chip finally result from a bad position of the trap in the picking process but can be solved by fixing the chip on the XYZ-stage. Anyway, the first approach is weak in terms of adaptation because a small displacement of the chip or a change of position of the traps in the chip would limit the sorting process. We aim to make the holding pipette able to access to an egg according to the pixel coordinates of the egg. Given that the field of view (FoV) doesn't change during all the process because the camera is fixed, only two coordinates of the micromanipulator are used during the picking process namely X and Y. The displacement of the micromanipulator for the picking process is in reality only two translational movements, under the x and y axes, while z axis is fixed for all the process. We create an equation to map camera pixel Y*, X* coordinates of the holdertip to global reference frame X, Y coordinates of the micromanipulator. To do so, we choose an initial position of the holdertip in the FoV and make travel the micromanipulator along the y axis throughout the FoV by taking images and recording the real positions of the manipulator corresponding to each image. We do the same operation for the x axis. These images are then applied to our YOLO model detection algorithm by adding a segmentaion algorithm named YOLOv5 Instance Segmentation for the holding pipette which will allow to recover the pixel coordinates of the holdertip (see Fig. 17(B)). We then plot the curves of the coordinates of the micromanipulator X , Y as a function of the pixel coordinates X* , Y* of the holdertip. The trend curves are thus recovered (see Fig. 16). We obtain a relation between the variation of the pixel coordinates of the holdertip and global frame coordinates of the micromanipulator. Assuming the micromanipulator is at the initial position (Xi,Yi) global reference frame and the holdertip is at (X*,Y*) pixel coordinate (see Fig. 17(A)). When a dead egg is detected by the model at (Xc,Yc) pixel coordinates, to minimize the distance $d$ (see Fig. 17) between the holdertip and the dead egg, it suffices to calculate the global reference frame coordinates of the manipulator defined by the mapping equations:

$$Xf = 0.0007Xc^2 + 2.3122Xc + 9141.9 \tag{8}$$

$$Yf = -0.0004Yc^2 + 2.0947Yc + 16227 \tag{9}$$

With the equations thus obtained, it is possible to move the holdertip in the FoV based on the pixel coordinates of the egg and tests show a margin of error of $\pm 2\mu m$, which is acceptable considering the size of zebrafish embryo (0.7 mm). We define the initial position of the micromanipulator the same as when establishing the mapping. In this mode, we move the stage continuously rather than in steps, and it only stops when a dead egg is found by the DL model. The step $'Bring\ the\ holding\ pipette\ close\ to\ the\ egg'$ of the algorigram (see Fig. 15(b)) is realized by putting the pixel coordinates of the detected embryo into the Eqs. 8 and 9. The

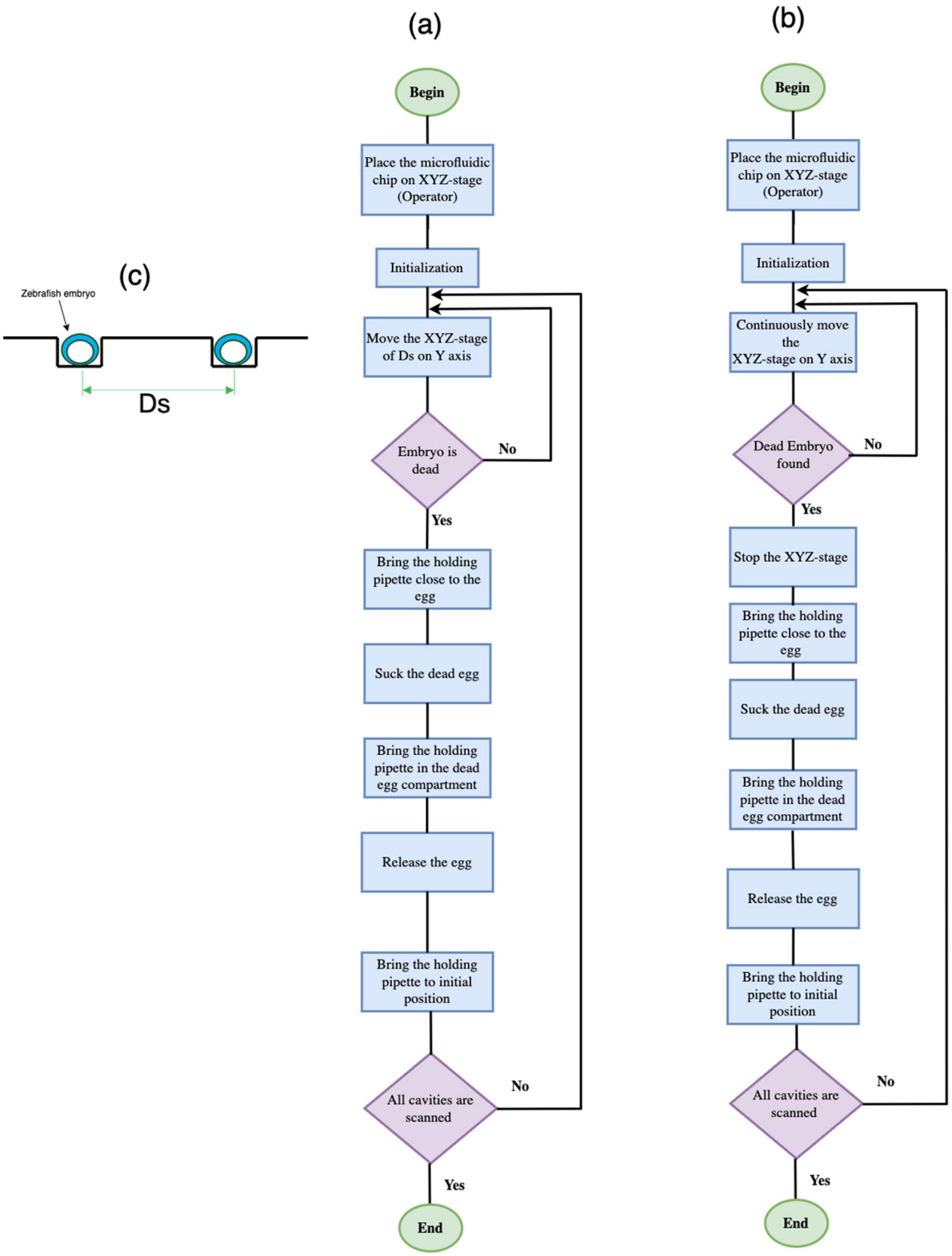


**Fig. 15** Algorigram of the robotic sorting system handling the case where only dead eggs are removed from the chip. (a) Without feedback use of dead egg position. (b) With feedback use of dead egg position. The steps $'Suck\ the\ dead\ egg'$ and $'Release\ the\ egg'$ are realized with a piezelectric pump at 500ml/min. The step : $'Bring\ the\ holding\ pipette\ in\ the\ dead\ egg\ compartment'$ means to reach the waste part of mirofluidic chip. (c) Illustration of distance Ds

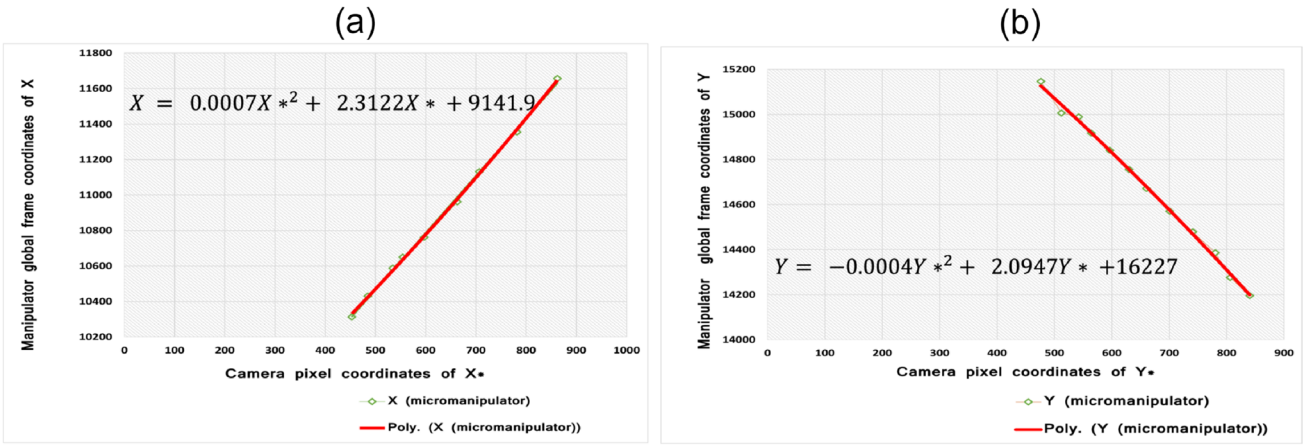


**Fig. 16** (a): Manipulator global frame coordinates X in micrometer evolution in the FoV according to the holdertip pixel coordinates X*. (b): Manipulator global frame coordinates Y in micrometer evolution in the FoV according to the holdertip pixel coordinates Y*. Reprinted with permission from IEEE, Copyright (2024), 5719300756547 [21]

two equations serve as a mapping tool that enables the holdertip to precisely locate and access dead eggs. In trials 3,4,5 and 6, the implementation of these formulas resulted in a 97% success rate for the pick and place process but a decrease of the embryo's detection accuracy which is 98%. This decrease in the accuracy of the classification of zebrafish embryos may be due to the movement of the stage which is not adequate with the number of frames that the camera can take because certain embryos will require more stability to be well seen. For example, the camera can capture frames with images of half-viewed embryos. By further reducing the speed of the XYZ stage, we will be able to obtain better results at the expense of the speed of the sorting process.

Sorting out dead or unfertilised zebrafish eggs is a crucial task for biologists. Our optimised detection model successfully detects zebrafish eggs with acceptable accuracy, depending on the operating mode. Additionally, it operates at high processing speed (10.6 ms per frame), enabling full exploration of the chip (36 egg traps) in 44 seconds, provided the traps contain viable eggs or are empty. The use of a microfluidic chip improves the process of accessing and handling eggs by reducing computation time and immobilizing the egg without contact, which can cause damage. The microfluidic chip enables eggs to be stored in cavities without being handled, which can also cause damage. The micromanipulator is calibrated without feedback using the dead egg position mode to fix the y and z axes. Knowing the distance between the cavities, the displacement of the XYZ-stage is chosen as being the distance which separates the cavities. Therefore, the cavity must be in front of the pipette holding and a translation on the x axis will allow the embryo to be recovered as shown in Fig. 18(X.B). However, a simple calibration error will upset the process without use of feedback for the dead embryo position. Using a feedback mode ie with feedback use of dead egg position, gives us more efficiency in the picking process but decrease the scanning speed of the chip. In continuous mode, the experiments made have shown that the maximum speed of the XYZ-stage which allows detection of zebrafish eggs is around 4 mm/s. Under these conditions, our microfluidic chip is fully explored in around 52 s if all of the traps are empty or contain livable eggs. The feedback mode offers more robustness because, a change of the microfluidic chip and its characteristics or a micro-displacement of the chip will not affect the process. The holding pipette no longer needs to be exactly in front of the embryo to access it. In Fig. 18(Y.B), the holding attacks the

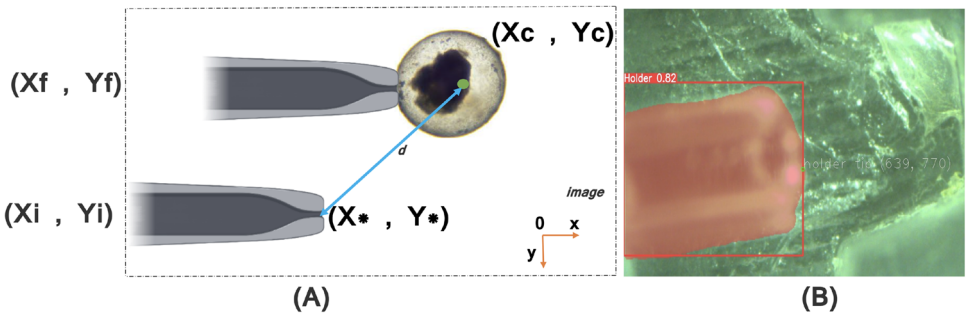


**Fig. 17** Illustration of pipette displacement. (Xf,Yf) : Micromanipulator real frame coordinates (final position); (Xi,Yi) : Micromanipulator real frame coordinates (initial position); (X*,Y*) : Pixel coordinates of the holdertip ; (Xc,Yc) : Pixel coordinates of the cell. (B) An example of detection followed by segmentation of the holding pipette

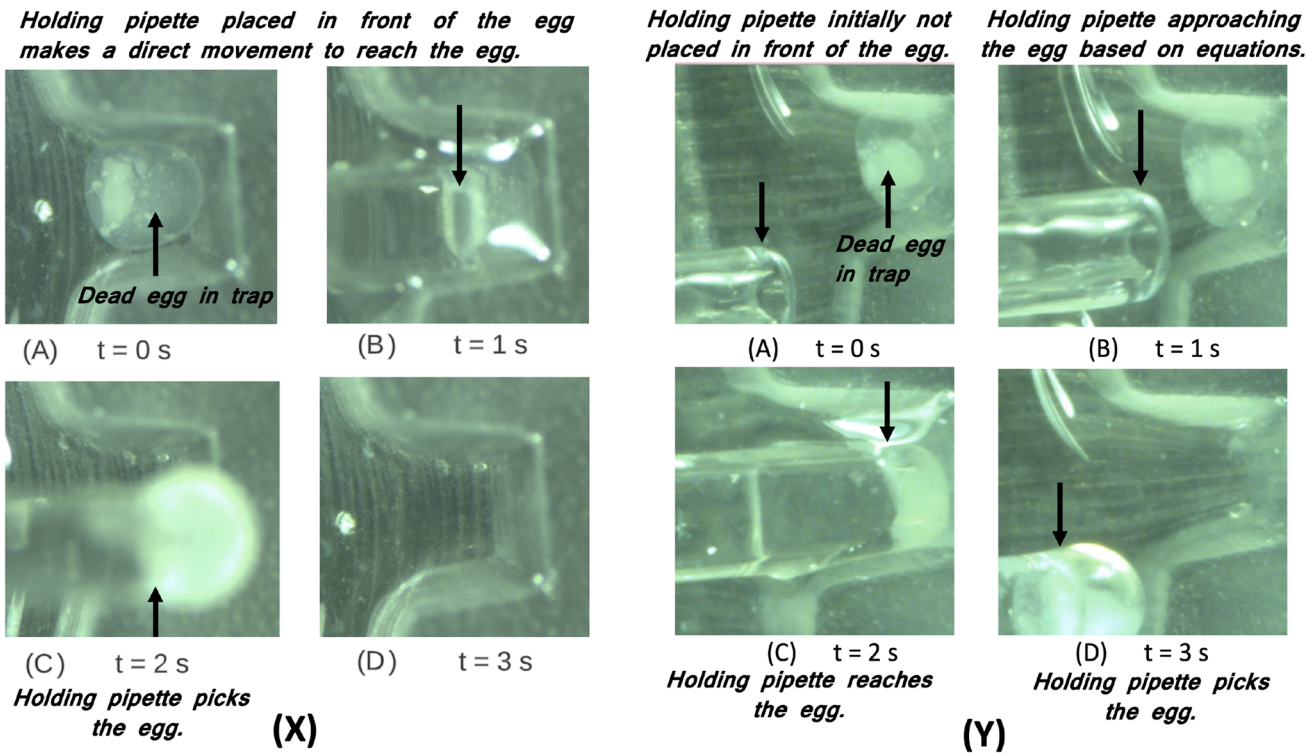


**Fig. 18** X: Without feedback use of dead egg position. (X.A): Dead egg in the trap, , holdertip no need to be in the FoV. (X.B): access of the pipette in the trap, aspiration in progress with 500ml/min during 1 s. (X.C): Egg already picked. (X.D): Empty trap, after aspiration. Y: With feedback use of dead egg position. (Y.A): Dead egg in the trap, holdertip in the FoV. (Y.B): Holdertip going to the egg position using map equations. (Y.C): Aspiration in progress with 500ml/min during 1 s. (Y.D): Egg already picked, empty trap after aspiration. When using feedback of dead egg position, the holdertip can be anywhere in the FoV and manages to access the egg compared to the other mode , the holdertip must be initialized to be in front of the starting trap and only moves along x axis to acess the dead egg. Reprinted with permission from IEEE, Copyright (2024), 5719300756547 [21]

cell from the side and manages to find the cell using the equations and pixel coordinates of the center of the embryo. This is a major advantage of the feedback mode because the accuracy of sorting increases but the speed of the exploration of the chip is reduced. This is normal because closed-loop systems work precisely due to the feedback system. Open loop system usually gives fast response, while closed loop system gives slow response. A more effective approach would be to combine a step-by-step movement of the XYZ stage with a high speed of 20 mm/s and use the map equations to identify the embryos to be sorted. To demonstrate the superiority of our proposed automated zebrafish embryo sorting system, we conducted a comparison with similar works in the literature.

Our sorting system (see Table 6) provides higher accuracy, egg detection and sorting speed compared to [8, 9]. However they have experimented with a larger number of zebrafish than ours. It is clear that nowadays the detection models are effective, the competition lies in the speed of execution of the task. Our work based on an optimized YOLOv5 model offers a much higher speed compared to the methods used in these two works, it is 94x faster than [9] based on template matching. Same observation for the speed of sorting an egg.

**Table 6** Comparison of our work with the literature

| Tasks | Our work | [9] | [8] |
|---|---|---|---|
| Num. of eggs for sorting experiments | 320 | 4752 | 694 |
| Egg sorting accuracy | 20% (trial 1)<br>90% (trial 2)<br>97.9% (trial 3,4,5 and 6 combined ) | - | 96.8% |
| Egg detection speed (per 1 egg) | 10.6 ms | 1 s | - |
| Egg sorting speed (per 1 egg) | 3 s | 14 s | 8 s |

In addition to the good immobilization and the sorting offered by the chip, it is believed that our approach may be useful in particular for rapid cell microinjection.

## 4 Conclusion

We have successfully developed a sorting system using the YOLOv5 deep learning detection algorithm to distinguish dead or unfertilised zebrafish eggs. After training a default model and then optimized by tuning hyperparameters of YOLOv5 through generations, our model achieves a remarkable speed of 10.6 milliseconds per frame with an accuracy of 95.8%. Compare to our previous work [21], the microfluidic chip developed for this work has a percentage of egg capture higher than the previous chip used. Likewise, the model acquires a $mAP@0.5$ and $mAP@0.5:0.95$ slightly higher than that used in the previous work. The optimized model was used for a sorting system composed of a microfluidic chip where the eggs are housed in cell traps, an XYZ-stage and a micromanipulator with a glass pipette as end-effector. Our experiments yielded a high sorting efficiency of 97.9% in feedback mode, surpassing similar systems. In the future, our focus will be on refining egg sorting using deep learning and microfluidic chip, aiming to eliminate external components like the micro-manipulator and XYZ-stage. Instead, we will aim to develop a portable sorting system using only water and pumping system.

**Acknowledgements** The authors thank Edouard Manzoni and Marco Amaral, technicians for the Animal Facility and Engineering Aquatic Models Platform of Sorbonne University for providing us fresh zebrafish eggs.

**Author Contributions** All authors have made substantial contributions regarding the conception and design of the study. Acquisition of images and analysis (AD, SH), labelling images (AD,AM,DZ), development of the deep learning model (AD, FS), Robotic phase sorting (AD, EG, MB), mechanical design and manufacturing of the microfluidic chip (AD, IF, GL). (AD,FS) wrote the main manuscript text and final approval of the version submitted (all authors).

**Funding** This study was funded by the Université franco-italienne (UFI) / Universitá Italo Francese (UIF).

**Data Availability** The datasets supporting in this study are available from the corresponding author upon reasonable request.

**Code availability** The codes used for zebrafish embryo detection and sorting in this study are publicly available at the GitHub repository: https://github.com/AliouneDiouf/zebappdetect.git

## Declarations

**Competing interest** The authors declare that they have no conflict of interest.

**Ethics approval and consent to participate** N/A

**Consent for publication** N/A

## References

1. Mohand Ousaid A, Haliyo S, Régnier S, Hayward V (2020) High fidelity force feedback facilitates manual injection in biological samples. IEEE Robot Autom Lett 5(2):1758–1763. https://doi.org/10.1109/LRA.2020.2969940
2. Woynarovich E, Horváth L et al (1980) The Artificial Propagation of Warm-water Finfishes: a Manual for Extension. vol 201
3. Meyers JR (2018) Zebrafish: development of a vertebrate model organism. Curr Protocols Essential Lab Tech 16(1):19
4. Otterstrom JJ, Lubin A, Payne E, Paran Y (2022) Technologies bringing young zebrafish from a niche field to the limelight. SLAS technology
5. LaBelle CA, Massaro A, Cortés-Llanos B, Sims CE, Allbritton NL (2021) Image-based live cell sorting. Trends Biotechnol 39(6):613–623
6. Sadak F, Saadat M, Hajıyavand AM (2020) A vision-guided methodology for the automation of biological cell injection. In: 2020 2nd International conference on electrical, control and instrumentation engineering (ICECIE), pp 1–9. IEEE
7. Schutera M, Dickmeis T, Mione M, Peravali R, Marcato D, Reischl M, Mikut R, Pylatiuk C (2016) Automated phenotype pattern recognition of zebrafish for high-throughput screening. Bioengineered 7(4):261–265
8. Graf SF (2011) Automated microinjection with integrated cell sorting, immobilization and collection. PhD thesis, ETH Zurich
9. Breitwieser H, Dickmeis T, Vogt M, Ferg M, Pylatiuk C (2018) Fully automated pipetting sorting system for different morphological phenotypes of zebrafish embryos. SLAS TECHNOLOGY: Translating Life Sci Innov 23(2):128–133
10. Shang S, Long L, Lin S, Cong F (2019) Automatic zebrafish egg phenotype recognition from bright-field microscopic images using deep convolutional neural network. Appl Sci 9(16):3362
11. Ishaq O, Sadanandan SK, Wählby C (2017) Deep fish: deep learning-based classification of zebrafish deformation for high-throughput screening. SLAS Discovery: Adv Life Sci R&D 22(1):102–107
12. Čapek D, Safroshkin M, Morales-Navarrete H, Toulany N, Arutyunov G, Kurzbach A, Bihler J, Hagauer J, Kick S, Jones F et al (2023) Embryonet: using deep learning to link embryonic phenotypes to signaling pathways. Nat Methods 1–9
13. Cordero-Maldonado ML, Perathoner S, Kolk K-J, Boland R, Heins-Marroquin U, Spaink HP, Meijer AH, Crawford AD, Sonneville J (2019) Deep learning image recognition enables efficient genome editing in zebrafish by automated injections. PLoS One 14(1):0202377
14. Jones RA, Renshaw MJ, Barry DJ, Smith JC (2023) Automated staging of zebrafish embryos using machine learning. Wellcome Open Res 7(275):275
15. Zhu F, Akagi J, Hall CJ, Crosier KE, Crosier PS, Delaage P, Wlodkowic D (2013) A high-throughput lab-on-a-chip interface for zebrafish embryo tests in drug discovery and ecotoxicology. In: Micro/Nano materials, devices, and systems, vol 8923, p 892345. International Society for Optics and Photonics
16. Lefevre A, Gauthier V, Gauthier M, Bolopion A (2022) Closed-Loop Control of Particles Based on Dielectrophoretic Actuation. IEEE/ASME Trans Mechatron 1:1–10
17. Choudhury D, Noort D, Iliescu C, Zheng B, Poon K-L, Korzh S, Korzh V, Yu H (2012) Fish and chips: a microfluidic perfu-

sion platform for monitoring zebrafish development. Lab Chip 12(5):892–900
18. Li Y, Yang F, Chen Z, Shi L, Zhang B, Pan J, Li X, Sun D, Yang H (2014) Zebrafish on a chip: a novel platform for real-time monitoring of drug-induced developmental toxicity. PLoS One 9(4):94792
19. Chen Z, Liu X, Tang X, Li Y, Liu D, Li Y, Huang Q, Arai T (2022) On-chip automatic trapping and rotating for zebrafish embryo injection. IEEE Robot Autom Lett 7(4):10850–10856
20. Tang X, Liu X, Li P, Liu F, Kojima M, Huang Q, Arai T (2020) On-chip cell–cell interaction monitoring at single-cell level by efficient immobilization of multiple cells in adjustable quantities. Anal Chem 92(17):11607–11616
21. Diouf A, Sadak F, Fassi I, Boudaoud M, Legnani G, Haliyo S (2023) Automatic sorting of zebrafish embryos using deep learning. In: 2023 International conference on manipulation, automation and robotics at small scales (MARSS), pp 1–6. IEEE
22. Westerfield M (2007) The zebrafish book; a guide for the laboratory use of zebrafish (danio rerio). (No Title)
23. Jocher G (2020) ultralytics/yolov5: V3.1 - Bug Fixes and Performance Improvements. https://doi.org/10.5281/zenodo.4154370

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

## Authors and Affiliations

**Alioune Diouf[1,2] · Ferhat Sadak[1,3] · Edison Gerena[1] · Abdelkrim Mannioui[5] · Daniela Zizioli[6] · Irene Fassi[4] · Mokrane Boudaoud[1] · Giovanni Legnani[2] · Sinan Haliyo[1]**

Ferhat Sadak
fsadak@bartin.edu.tr

Edison Gerena
edison.gerena@isir.upmc.fr

Abdelkrim Mannioui
abdelkrim.mannioui@upmc.fr

Daniela Zizioli
daniela.zizioli@unibs.it

Irene Fassi
irene.Fassi@stiima.cnr.it

Mokrane Boudaoud
mokrane.boudaoud@isir.upmc.fr

Giovanni Legnani
giovanni.legnani@unibs.it

Sinan Haliyo
sinan.haliyo@isir.upmc.fr

1 Sorbonne Université, Institut des Systèmes Intelligents et de Robotique (ISIR), Paris, France

2 Department of Industrial Engineering, Universitá degli Studi di Brescia, Brescia, Italy

3 Department of Mechanical Engineering, Bartin University, Bartin, Türkiye

4 Institute of Intelligent Industrial Technologies and Systems for Advanced Manufacturing National Research Council of Italy, Milan, Italy

5 Sorbonne Université, Institut de Biologie Paris-Seine (IBPS), Paris, France

6 Department of Molecular and Translational Medicine, Universitá degli Studi di Brescia, Brescia, Italy