Blending in with the Shoal: Robotic Fish Swarms for Investigating Strategies of Group Formation in Guppies

Tim Landgraf[1], Hai Nguyen[1], Joseph Schröer[1], Angelika Szengel[1], Romain J.G. Clément[2], David Bierbach[2], and Jens Krause[2]

[1] Freie Universität Berlin, FB Mathematik u. Informatik
Arnimallee 7, 14195 Berlin, Germany
[2] Leibniz-Institute of Freshwater Ecology & Inland Fisheries
Müggelseedamm 310, 12587 Berlin, Germany
tim.landgraf@fu-berlin.de
http://biorobotics.mi.fu-berlin.de

Abstract. Robotic fish that dynamically interact with live fish shoals dramatically augment the toolset of behavioral biologists. We have developed a system of biomimetic fish for the investigation of collective behavior in Guppies and similarly small fish. This contribution presents full implementation details of the system and promising experimental results. Over long durations our robots are able to integrate themselves into shoals or recruit the group to exposed locations that are usually avoided. This system is the first open-source project for both software and hardware components and is supposed to facilitate research in the emerging field of bio-hybrid societies.

Keywords: biomimetic robots, biomimetics, swarm intelligence, social behavior, social networks, swarm tracking.

1 Introduction

The study of collective animal behavior can benefit substantially from the use of biomimetic robots. Once accepted by the animal group as a conspecific, the robots can be used to test theoretical models of group formation, leadership, mate choice and other biological functions and mechanisms. In recent years this approach has been shown in various animal models, such as in cockroach shelter seeking [1], honeybee dance communication [2], bowerbird courtship behavior [3] and decision making in fish [4]. While the complexity of those robotic systems is still low in number of actuators and behavioral repertoire (for a review see [5] or [6]), a few interactive, closed-loop systems for robotic fish have been proposed only recently [7–9]. Those systems all use tracking devices that visually recognize the individuals of the shoal. This information is fed back to the control of the robots and therewith allows a dynamic interplay between robots and animals. However, significant biological findings, obtained with such interactive systems, have yet to be put forward. In interactive systems, the robots have to

A. Duff et al. (eds.): Living Machines 2014, LNAI 8608, pp. 178–189, 2014.

reach group acceptance with proper appearance but also on the level of social behavior. The integration of artificial agents therefore is a fragile process and can only be sustained with robust tracking systems, a finely tuned set of motion controllers and carefully composed robotic behaviors. In this contribution we present a detailed description of our interactive system for robotic fish, soon be published as open-source software and hardware specifications. We hope that this will facilitate research in this exciting new field of bio-hybrid systems. This paper is divided into five parts: 1) general system description, 2) computer vision implementation for tracking the robots and live fish, 3) description of interactive behaviors 4) experimental validation and results and 5) conclusions.

2 General System Description

Small shoals of up to 20 full-grown *Guppies* are kept in a water tank of 1 m^2 floor area filled with only 15 cm of water. The tank is positioned at about 1.40 m above the ground. Two-wheeled differential drive robots are moving below the tank on a transparent platform (Figure 3). Each of the robots holds up a neodymium magnet to the bottom side of the tank in which fish replicas, magnetically coupled, follow the robots' movements (see Figure 1). The artificial fish are moulded using dead template animals, painted and finished for a realistic morphology and appearance. The robots carry two infrared LEDs on their bottom sides. On the ground, a camera with an IR-pass filter glass is facing upwards to track the movements of the robots. A second camera is affixed above the tank to track both real and artificial animals within the tank. One computer is running a program for tracking the robots and sending motion commands to each individual robot over a wireless channel. A second computer evaluates the video feed of the shoal camera and sends tracking data to the first computer via a local area network connection.

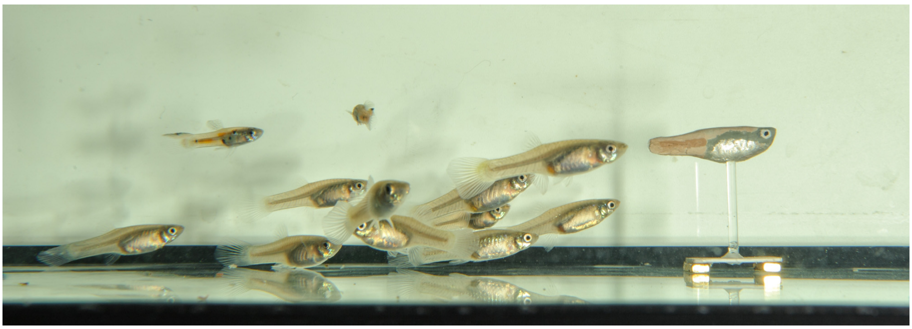

Fig. 1. The replica is attached to the magnetic base with a thin transparent plastic stick of 1 mm width. The photograph shows a Guppy replica ahead of a swarm of 12 animals.

2.1 Robot Design and Control

The two-wheeled robot's aluminium frame and the wheels are custom built with a base area of 7 $cm \times 7$ cm as depicted in Figure 2. The main electronics are three Arduino-compatible boards (so called "shields") for main processing (Arduino Uno), WiFi communication (Copperhead WifiShield) and motor control (DFRobot Motor Shield). We use a two-cell LiPo battery pack (7.4 V nominal output) that connects to a voltage regulator (Fairchild Semiconductor LM350T) and a voltage divider.

The former provides a constant voltage of 5 V with currents of up to 3 A irrespective of the battery level. The regulator's output is fed to the Arduino main board and the motor shield which generates PWM signals for each motor from this constant input voltage. The voltage divider scales down the battery output which then can be measured on an analog pin of the Arduino board with respect to an internal reference (1.1 V). Two Faulhaber DC gear motors are affixed to the side plates and directly connect to the wheels that we produce from circular plates into which V-shaped notches are turned. Rubber gasket rings fitted in the notches serve as treads for traction. Each robot carries a strong rare-earth magnet at the end of a plastic rod held up against the tank's floor without physical contact. The tank itself is made of glass that due to the mass of the water column bends slightly downwards. The distance of the magnet to the glass is therefor less than 1 mm near the center of the tank and approx. 2 mm in the periphery. The magnet's poles are aligned in parallel to the running surface.

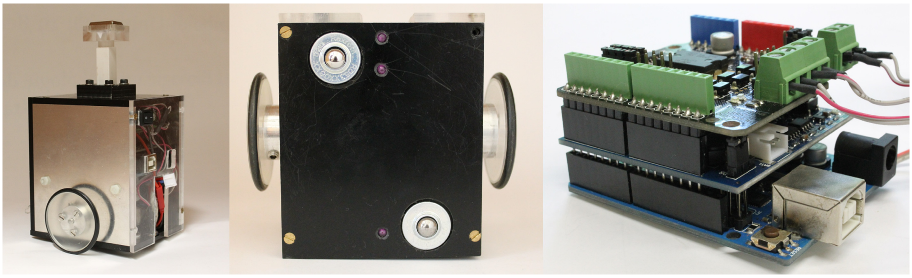

Fig. 2. We use two-wheeled differential drive robots whose frame is made of aluminium and plastic. A rare-earth magnet is attached to the robot's top side, adjustable in height. The robot is moved by two DC gear-motors that directly connect to two wheels. Two ball bearings in the front and back are used for stability. Three IR-LEDs in alignment with the robots forward direction are inset into the base plate. A stack of Arduino-compatible boards is used for motor control and wifi communication.

The control program for the agents is executed on a personal computer. The main control loop is run at a frequency of 30 Hz as determined by the bottom camera's frame rate. In each time step, a command packet is issued and sent to the individual robots via WiFi (UDP). Each robot has a unique IP-address and only receives its respective packets. We use a fixed length protocol with a two bytes header, 12 bytes data and two bytes checksum. The motion command consists of two motor speeds as computed by a PID controller based on the current robot's position and future target points. Targets can be defined as a static sequence or, dynamically, by integrating the output of the shoal tracking. The proportional component of the controller is defined by two sigmoidal functions, that define the forward and turning speeds over the Euclidean and angular distances to the target point, respectively. Once a robot has reached its next target position, the next location in the sequence is selected as the new target. In interactive behaviors, properties of the shoal or its members are used to define target points on the fly for every new time step. Depending on the starting conditions and the shoal's behavior, this might result in very different trajectories. In section 4 we describe the interactive behaviors in more detail. The firmware on the robots receives WiFi packets and translates the data content. Currently, the robots implement the following sub-routines:

1. Sending status packets back to the control computer at 1 Hz, including the robot's unique identifier and battery level
2. Toggling of infra-red LEDs for the identification of robot
3. Generating PWM signals according to the received motor speed values
4. Toggling the so-called burst mode, in which slow motor speeds are translated to PWM signals that are preceded by short bursts of a higher frequency to overcome initial friction and inertia.

3 Computer Vision

We propose using two sensory channels: one for controlling the robotic agents and one for measuring social interactions within the tank. Especially in dense groups, robots might get confused with real fish when using just one tracking system from above. This is a major difference to the systems proposed by [8] and [7].

3.1 Robot Tracking

We propose attaching three IR-LEDs to the bottom sides of the robots. When operated on a transparent platform, each robot can be easily localized in the video feed of a camera on the ground. For this task, we utilize a standard webcam (Logitech Pro 9000) with the IR-block cover glass replaced by an IR-pass filter. Hence, the infrared LEDs produce very bright spots in the otherwise dark camera image. Hence, the computer vision pipeline is designed minimalistically:

First, a global threshold is applied to binarize the image, which is then denoised by an erosion operation. Possible remaining gaps in the blobs are filled by applying a dilation. Subsequently, we seek connected components that likely represent IR-LEDs. Since the distance of the LEDs on each robot is known and constant, the distance of the blobs in the image is known and constant as well.

Fig. 3. The wheeled robots are moving on a transparent plastic plate in a space below the water tank. A camera on the floor is used to track the individual robots that each have three infrared LEDs inset in their baseplate.

Two of the three LEDs are placed in close vicinity, such that their resulting blobs merge to a significantly larger object than the one resulting from the single LED. Using the distance constraint we identify pairs of heterogeneously sized blobs that then define the location and orientation of each robot. Since robots might come close to each other we additionally check whether the candidate blob pair is not too far away from previous detections. Before the robots are operated, the program initially broadcasts a request to all available robots to report their IDs. Then, sequentially, each robot is requested to blink its LEDs. The vision system detects this event and assigns the respective ID to the formerly unidentified virtual object in the tracking arena. For all following frames new object detections are assigned the ID of the nearest previously known robot locations. All coordinates are rectified and translated to the world coordinate system using a user-defined rectangle that matches the arena's outline in the camera projection.

3.2 Fish Tracking

All interactions of robots and live fish are observed from above the tank by a second camera. In order to detect all individuals we use a background subtraction procedure that models foreground and background pixel distributions with a mixture of Gaussians [10]. Since the tank's bottom and walls are laminated with white plastic the fish appear as clear dark objects in the camera image. Once converged, the background model shows an empty tank such that

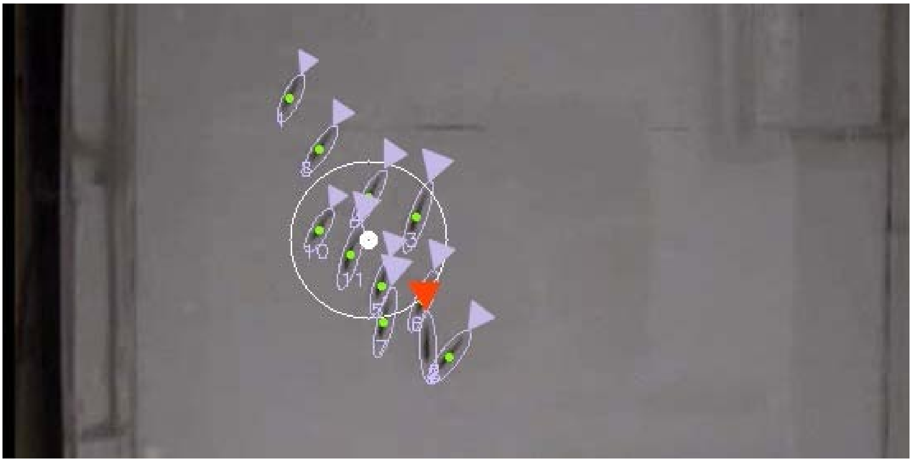

Fig. 4. A (cropped) screenshot of the shoal tracking system. Each individual of the shoal is identified by a number. An ellipse and a triangular tail is depicted to mark each fish's position. The orange tail marks the only robot in this recording. The green dots signify that those animals have been found to be in one subgroup, whose center is denoted by the white dot. The circle around the dot denotes one standard deviation.

we get clear positive peaks in the difference image of background and current live frame. The individual fish are detected by applying a global threshold to the difference image. All regions having above-threshold values are used for seeding a customized multi-agent particle filter (unpublished, based on [11]). Each of the 500 particles, representing a point in state space, is scored using the difference image: A particle is defined by its world coordinates, an orientation angle and length and width. A given parameter combination defines an ellipse in image space. The match to the image is computed as the difference of the sum of all pixels within the ellipse and the sum of pixels in a ring around the ellipse.

The particle filter uses importance resampling to select a number of particles that by this process iteratively converge to the optimal locations, as defined by the matching function. Per frame, after a number of resampling iterations, clusters of particles are averaged to obtain a robust location estimate for each individual. The fishs' orientation angles cannot be robustly obtained on still images. Since fishes usually swim forwards, the direction is disambiguated by integrating the motion vector over a fixed time window. The system assigns an ID to every fish object, i.e. a cluster of particles in state space, and tracks them using a simple model of fish motion. For a given cluster center, its motion speed in three-dimensions (planar position and orientation) defines an expected position for future frames. We distribute new hypotheses along this motion axis using Gaussians centered at the measured velocities. Since the number of fishes in the tank is known and constant we keep an according number of cluster representatives and update their locations iteratively over time. Figure 4 depicts a sample shoal with overlaid tracking information. The shoal tracking system was validated in two regards: first, the positional error of static objects was calculated by placing fish-sized metal blocks on known positions and comparing the system's output to the reference positions. The average error is below 1 mm (std: 2 mm) but might be larger when objects are moving or the polygon that defines the homography is set erroneously. Secondly, the tracking error was determined by counting the number of individuals that were either lost by the system or assigned the wrong ID (e.g. after swimming close to another individual). If an individual gets lost and found again, we would only count the loss. Five different video sequences of the same duration (1000 frames, 40 seconds) and varying number of individuals were subject to the tracker. In average 1.2 errors per minute occur - most of them produced in one sequence with many fishes overlapping.

4 Interactive Behaviors

Interactive behaviors use the continuous stream of sensory feedback from the fish tracking system. For each frame a TCP-packet containing individual positions and cluster information to the robot control computer. There, a callback routine is triggered that collects the data and makes it available in a respective data structure that holds the robots' positions as well as provided by the robot tracking component. We have defined atomic behaviors, such as following a certain individual, going to a location in the tank, waiting, exploring, wall following, alignment with individuals, and so on. Atomic behaviors are preconfigured with appropriate values for parameters such as motion velocities, distance constraints with regard to other individuals and temporal parameters such as typical waiting durations. Each interactive behavior is implemented as an exchangeable module that logically assembles atomic behaviors dependent on the actual situation. All interactive behavior modules exhibit a generic structure. They can access the current set of world information (positions and orientations of all fishes and robots) and produce a target point or motor speeds directly for every time step. For each robotic agent a different behavior can be defined. For example, one agent can be set to follow another robot for all times and this one can be configured to follow a live fish. We have implemented a number of complex behaviors:

Follow Swarm Center. This behavior finds clusters of fish and chooses either the largest or closest cluster to follow. A target point is produced that either lies in the middle of the swarm, or, if the swarm is very dense, on the cluster bounds. If no clusters can be found, the nearest neighbor is selected to be followed. Collision avoidance is implemented rudimentarily: If individuals are likely being hit by the replica, the robot slows down and continues its way to the target leaving the potential evasive maneuver to the fish.

Swarm Integration Behaviors. We have implemented a behavior to mimic schooling inspired by the model of Couzin and coworkers [12] that describes fish schooling behavior as a simple sequence of attraction, alignment and

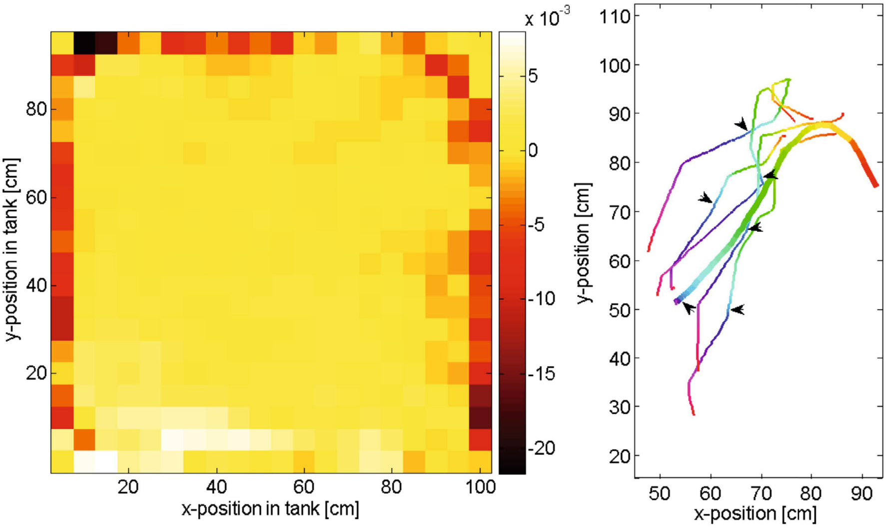

Fig. 5. Left: Difference image of animals' probability of presence in interactive recruitment experiments and reference scenario. All red and orange squares denote locations that were visited less often in the recruitment scenario. White and yellow squares, predominantly in the center of the tank, denote regions with higher occurence compared to a no-robot scenario. Right: Animals often copy the trajectory of the robot. The figure shows the movements of five live individuals and the path described by the robot (thick line). Time is color-coded and ranges from red over blue to violet for the duration of 7 seconds. The arrows denote the positions of all individuals at the time point the robot stopped near the target point. All animals follow after the robot took the lead towards the center of the tank. Intriguingly, most individuals seem to copy the direction of the robot more than half a second after the robot initiated its move.

repulsion behaviors. Therefore, three concentric zones are defined around the robot. If e.g. the agent measures another individual within its repulsion zone, a target point away from that animal is produced. If a fish is swimming in the alignment zone, the robot is rotated in order to match the orientation of that animal. If an animal is in the attraction zone, it is approached. The rejection rule has priority over the other rules; if there are any fish present in the rejection zone, only the rejection maneuver will be executed. If there are no individuals within the rejection zone, but both the alignment and attraction zones contain individuals, the robot will use the average of the motion vectors resulting from the alignment and attraction rules.

Predator Behavior. The predator behavior is a variant of *Follow Swarm Center*. The motion speeds are higher and the target point is always the center of the swarm. Minimal distances to individuals are ignored.

Recruitment Behaviors. There are many combinations of atomic behaviors and parameters that make up recruitment behaviors. All basically consist of two stages, an approach and swarm integration part, and the recruitment phase. In the behavior used for the experiments the robot executes the behavior *Follow Swarm Center* when far away from the shoal (defined parametrically). Second, when the robot has spent a certain duration with or inside the swarm, the center of the tank is set as the new target point. Following further options for target points are already implemented: a) a fixed location anywhere in the tank or a location drawn from a normal distribution centered at a given spot, b) a random point in the vicinity of the robot, c) a random point that lies ahead of the robot, using polar coordinates drawn from a two normal distributions centered on the agent's current direction and a distance of 10 cm. For the experiments described below, we configured the robot to wait for five seconds once the target point is reached.

5 Experiments

In order to quantify how well robotic fish are accepted by the group, one may look at a variety of behavioral parameters. The spatial resolution of our camera system yields approximately 10 pixels per fish only. Hence, our analysis focuses on body positions and derived parameters only. In this paper we compare how fish are distributed in different scenarios. As a reference, a natural shoal, without robots was kept in the water tank for 30 minutes and was video recorded. In the second experiment, the robot described a static square-shaped trajectory for the same time. In the third experiment, the robot was configured to follow the shoal's center and, in the last experiment, executed the recruitment behavior for luring the animals to the center of the water tank, a location that is usually avoided. The positions and orientations were extracted and the tracking data was evaluated subsequently in MATLAB. We generated a 2-dimensional histogram of the fishs' positions which is normalized by the total number of occurences. This results in a map of presence probability for any fish over the entire tank area. Furthermore, we calculate the average inter-individual distance over all N animals for each time step t as:

$$d_{II}(t) = \frac{2}{N(N-1)} \sum_{j=1}^{N} \sum_{\substack{i=j+1,\\ i \neq j}}^{N} \|\boldsymbol{p}_i(t) - \boldsymbol{p}_j(t)\|_2$$

where $\boldsymbol{p_i(t)}$ is the planar position of the i-th animal at time t. This measure grows larger when the shoal spreads, even when similarly dense sub-shoals are formed. To capture the general groups cohesion we calculate the average nearest-neighbor distance as:

$$d_{NN}(t) = \frac{2}{N(N-1)} \sum_{j=1}^{N} \min_{\substack{i=1..N,\\ i \neq j}} \|\boldsymbol{p}_i(t) - \boldsymbol{p}_j(t)\|_2$$

This is done for the no-robot scenario as well as for the mixed groups scenarios, ignoring the positions of the robot in all cases. In the following we characterize the natural, no-robot scenario and show how robots influence the shoal's behavior.

5.1 Results

Natural shoals tend to first explore the tank and then stay in one of the four corners most of the remaining time of the test. There are individual differences in the frequency of exploratory behaviors. In one no-robot experiment, two of the seven animals frequently left the group alone, individually. This results in a broader distribution of inter-individual and nearest-neighbor distances. The animals in the other experiments were less bold, reflected by a narrower distribution. The 2-dimensional distribution of the probability of presence exhibits high peaks at the corners and almost no fish near the center of the tank. In two independent trials of 30 minutes duration animals occupied a region of 50 cm x 50 cm centered on the middle of the tank in only 5 % of the time. A robot running on a static square-shaped trajectory does occasionally attract animals. However, the following behavior (Figure 5 depicts a sample run) is often interrupted after only a few seconds. In less than 8 % of the time animals are present in the inner square. Operating a robot with the interactive recruitment behavior described above yields a presence in the inner square in more than 15 % of the time. Animals that were recruited by the robot are staying for a short while in the inner

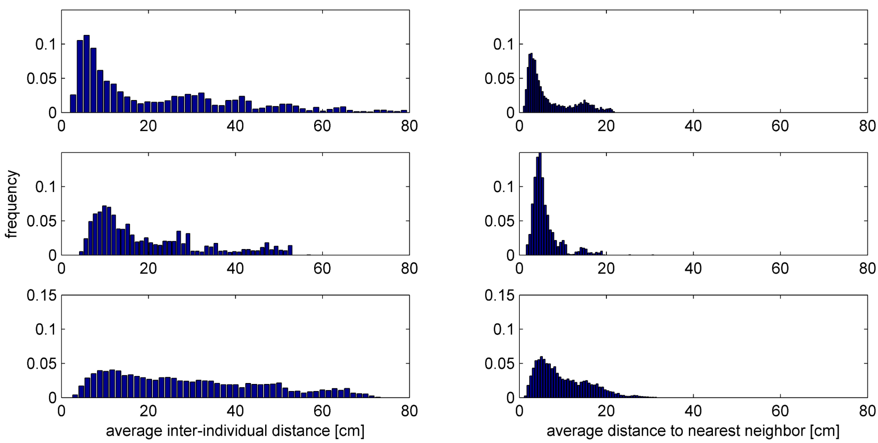

Fig. 6. Distributions of average inter-individual distance (left column) and average distance to nearest neighbor (right column) for three scenarios: the upper row shows the distributions of the reference case with no robots. The middle row depicts very similar distributions measured with one robot following the shoal's center. The recruitment behavior as shown in the last row exhibits a broader distribution. Note that in the two latter cases the robots have been excluded in the computation of the distance averages to reflect the shoal-only behavior.

square but then shortly swim off to the periphery, presumably because the robot stopped and stood still once it reached the tanks center. Example videos of the experiments can be found at [13]. Figure 5 depicts the difference of the distributions of the no-robot scenario and the recruitment behavior. Generally, each individual in the shoal seems anxious to stay with the group. Animals that explore the tank on their own can be observed only rarely. This is expressed in the distributions of average inter-individual and nearest-neighbor distances. Figure 6 shows the distributions of both measures over full experiments. Remarkably, a simple swarm following behavior produces a distributions resembling those of the reference case. The interactive recruitment behavior attracts single or a few individuals only, which is reflected in broader distributions, similar to the experiment with bold animals. However, Kolmogorov-Smirnov tests for equality of distributions yield no significant similarities for all pairs of distribution combinations.

6 Conclusions

We have developed a low-cost multi-agent platform for biomimetic robotic fish that resemble in appearance and behavior their live counterparts. The system allows dynamic interactions with single individuals or groups of fish using exchangeable modules of behavioral logic. The system is scalable for using many other agents under even larger tanks. However, due to the size of the wheeled robot under the tank, the density of robotic shoals is limited. The tracking of individual fish is robust, though it should be further improved in future research, e.g. by using cameras with higher spatial resolution. This would improve the quality of upcoming biological research by allowing smaller (i.e. younger) guppies to be used in experiments in which age distribution affects the focal behavior. The experiments indicate that closing the feedback loop enables the robot to display natural behavior expressed in similar distributions of inter-individual and nearest-neighbor distances. By applying a simple recruitment behavior, a single robot was able to lure away a few individuals from the shoal. However, the behavioral parameters, such as motions speeds and minimal distances to other individuals might still be chosen sub-optimally. In a few occasions, the robot seems to come too close to the fish or move too fast within the shoal. The affected individuals quickly evade the robot. In part, our results reflect these occasions in the distance distributions as well as in the spatial presence distribution. By opening our code base to the public, we invite other groups to join this specific research line and hope that this will facilitate research in bio-hybrid systems in general. Currently, we are preparing experiments with live *Guppies* and two robotic fish to investigate group decision making. For example, depending on the individuals' different experiences previously made with robots which were able or unable to find food. In addition, we plan to test this system with an implementation of a variety of theoretical models for schooling, mate choice and other biological functions in order to investigate whether the simulated results match our observations.

References

1. Halloy, J., Sempo, G., Caprari, G., Rivault, C., Asadpour, M., Tache, F., Said, I., Durier, V., Canonge, S., Ame, J., et al.: Social integration of robots into groups of cockroaches to control self-organized choices. Science 318(5853), 1155–1158 (2007)
2. Landgraf, T., Oertel, M., Rhiel, D., Rojas, R.: A biomimetic honeybee robot for the analysis of the honeybee dance communication system. In: IROS, pp. 3097–3102 (2010)
3. Patricelli, G.L., Uy, J.A.C., Walsh, G., Borgia, G.: Sexual selection: male displays adjusted to female's response. Nature 415(6869), 279–280 (2002)
4. Faria, J.J., Dyer, J.R., Clément, R.O., Couzin, I.D., Holt, N., Ward, A.J., Waters, D., Krause, J.: A novel method for investigating the collective behaviour of fish: introducing 'robofish'. Behavioral Ecology and Sociobiology 64(8), 1211–1218 (2010)
5. Krause, J., Winfield, A.F., Deneubourg, J.L.: Interactive robots in experimental biology. Trends in Ecology & Evolution 26(7), 369–375 (2011)
6. Mitri, S., Wischmann, S., Floreano, D., Keller, L.: Using robots to understand social behaviour. Biological Reviews 88(1), 31–39 (2013)
7. Bonnet, F., Rétornaz, P., Halloy, J., Gribovskiy, A., Mondada, F.: Development of a mobile robot to study the collective behavior of zebrafish. In: 2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), pp. 437–442. IEEE (2012)
8. Swain, D.T., Couzin, I.D., Leonard, N.E.: Real-time feedback-controlled robotic fish for behavioral experiments with fish schools. Proceedings of the IEEE 100(1), 150–163 (2012)
9. Landgraf, T., Nguyen, H., Forgo, S., Schneider, J., Schröer, J., Krüger, C., Matzke, H., Clément, R.O., Krause, J., Rojas, R.: Interactive robotic fish for the analysis of swarm behavior. In: Tan, Y., Shi, Y., Mo, H. (eds.) ICSI 2013, Part I. LNCS, vol. 7928, pp. 1–10. Springer, Heidelberg (2013)
10. Zivkovic, Z., van der Heijden, F.: Recursive unsupervised learning of finite mixture models. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(5), 651–656 (2004)
11. Isard, M., Blake, A.: Condensation conditional density propagation for visual tracking. International Journal of Computer Vision 29(1), 5–28 (1998)
12. Couzin, I.D., Krause, J., James, R., Ruxton, G.D., Franks, N.R.: Collective memory and spatial sorting in animal groups. Journal of Theoretical Biology 218(1), 1–11 (2002)
13. Landgraf, T., Nguyen, H., Schröer, J., Szengel, A.: Robo Fish LM 2014 Videos (2014), http://robofish.mi.fu-berlin.de/videos/lm2014