

Received 2 July 2026, accepted 21 August 2026, date of publication 1 September 2026, date of current version 8 September 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3729374

RESEARCH ARTICLE

Robust Motion Tracking and Classification of Zebrafish With Deep Learning

MAXIMILIAN NEIDHARDT^{1,2,3}, DEBAYAN BHATTACHARYA¹, IMKE A. K. FIEDLER^{2,4,5},
SANDIPAN MUKHERJEE¹, NICO FISCHER^{1,4}, BJÖRN BUSSE^{2,4,5},
AND ALEXANDER SCHLAEFER^{1,2,3,5}, (Senior Member, IEEE)

¹Institute of Medical Technology and Intelligent Systems, Hamburg University of Technology, 21073 Hamburg, Germany

²Interdisciplinary Competence Center for Interface Research (ICCIR), 20251 Hamburg, Germany

³SustAInLivWork Center of Excellence, 44249 Kaunas, Lithuania

⁴Department of Osteology and Biomechanics, University Medical Center Hamburg-Eppendorf, 20251 Hamburg, Germany

⁵RTG 3144 “Multiscale Imaging and Analytics of Interfaces in Musculoskeletal Health,” 20251 Hamburg, Germany

Corresponding author: Maximilian Neidhardt (Maximilian.Neidhardt@tuhh.de)

This work was supported in part by the Interdisciplinary Competence Center for Interface Research (ICCIR) through Hamburg University of Technology (TUHH) and University Medical Center Hamburg-Eppendorf (UKE), TUHH i^3 Initiative, the RTG 3144 “Multiscale Imaging and Analytics of Interfaces in Musculoskeletal Health” funded by Deutsche Forschungsgemeinschaft (DFG) under Grant 547468385, in part by the Forschungszentrum Medizintechnik Hamburg (FMTHH) Funding Program under Grant 03f1mthh2018, and in part by the Open Access Publishing Program of TUHH.

ABSTRACT Zebrafish serve as a key pre-clinical model for disease research. Studying disease progression requires long-term tracking of fish motion, particularly when considering the function of the musculoskeletal apparatus. While vision-based tracking has been studied, current approaches typically require substantial fine tuning and often lack robustness, especially when multiple fish need to be tracked and motion occurs along all three spatial dimensions. We propose a simplified setup and a 3D tracking method using only two cameras and two convolutional neural networks (CNN) to localize and track multiple fish robustly. We separate the localization of fish and their classification as an individual. This allows for simple training data acquisition by placing fish into the basin individually, without the need for manual annotation. We evaluate the approach on sets of two, five, and ten fish resulting in multiple object tracking accuracies (MOTA) of 0.997, 0.994, and 0.966, respectively. The mean MOTA over all scenarios is 0.99 compared to a mean MOTA of 0.89 for a state-of-the-art tracking algorithm (*idtracker.ai*). Finally, we present comprehensive data on the motion activity of individual fish in a 30-minute video sequence, demonstrating robust tracking and unique activity profiles of each fish.

INDEX TERMS Deep learning, fish tracking, fish trajectory dataset, locomotion pattern, physical activity.

I. INTRODUCTION

Zebrafish are widely recognized as a prominent model organism in biomedical research. Compared to other animal models, zebrafish allow for high-throughput genetic manipulation and the generation of large numbers of offspring [1]. Their suitability for studying the dynamics of the musculoskeletal system and how it can be affected by disease conditions has been increasingly acknowledged [2], [3], [4], [5], [6]. Due to the small body size of zebrafish, it is possible to efficiently evaluate disease progression and rehabilitation at high efficiency [7]. It is suggested that disorders affecting

musculoskeletal function in the zebrafish model may exhibit distinct patterns of physical activity and locomotion, e.g., swimming speed, agility, and maneuverability, stemming from pathological changes in their skeletal structures and tissues [8], [9]. Thus, the physical activity and locomotion patterns of individual fish can correlate with disease severity. Hence, the identification of individual fish within a group and the long-term recording of quantitative motion activity is strongly desired, as this allows correlating motion sequences with musculoskeletal body markers, e.g., bone mass, muscle mass, and morphology [10].

A widely used approach is camera-based tracking, which has also been employed to study the collective behavior of zebrafish [11], [12]. Typically, multiple cameras are used and

The associate editor coordinating the review of this manuscript and approving it for publication was Jeon Gwanggil.

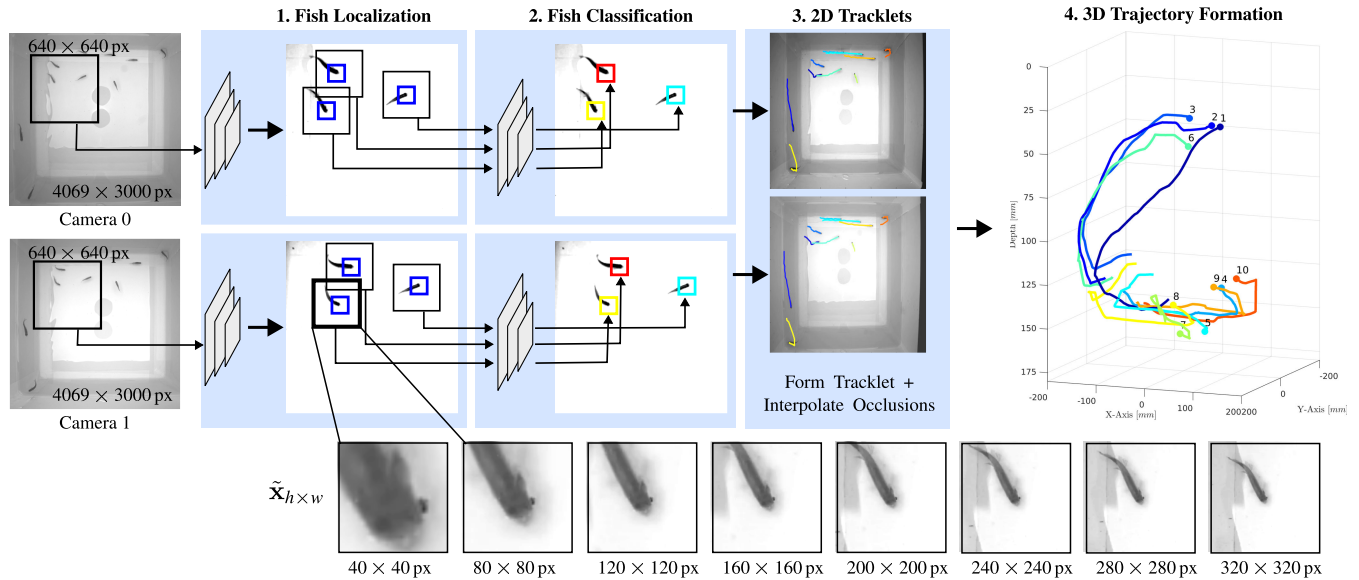


FIGURE 1. Data processing: The two camera streams are processed separately. (1) The location of all fish in the image frame is estimated by a fish localization model O . Patches $\tilde{x}$ of size $h \times w$ px are cropped with the fish head positioned at the center. Example patch sizes $\tilde{x}$ are depicted at the bottom. (2) Patches are classified by a classification model C . (3) 2D tracklets are estimated. (4) 3D trajectories are established through mid-point triangulation.

tracking consists of localizing and identifying individual fish in each camera frame, matching individual fish over time, and establishing the 3D trajectory by triangulation. Tracking approaches on short sequences have been demonstrated with conventional image processing methods, feature point extraction [13], [14], feature descriptors [15], depth-resolved cameras [16], adaptive segmentation algorithms [17], or blob-contour identification [18]. However, these methods rely partially on hand-crafted parameters and can be error-prone, particularly in long-term tracking scenarios, e.g., due to rapid body movements and frequent occlusions when multiple fish are studied.

More recently, convolutional neural networks (CNN) have been considered to detect and track zebrafish motion [12], [14], [19], [20], [21]. Combining CNNs with conventional image processing techniques has shown potential in 2D tracking scenarios. For example, a conventional blob detection algorithm localizing the fish head has been combined with a CNN for matching sequences of fish trajectories to individual fish [20]. Also, object detector models have been studied for fish detection from RGB images [12], [22], [23], [24], [25]. Following, similarities between fish patches are estimated to form trajectories of individual fish. Essentially, a similarity score is computed from handcrafted features or CNN based feature extractors [12]. Also, YOLO models have been studied for efficient fish detection [19], [26] and applied to classify different fish species [27], [28]. Also, transformer based models have been studied for tracking larger fish with complex background disturbance [29], [30]. The authors of *idtracker.ai* [21] propose a conventional blob detection algorithm to localize individuals, which are subsequently classified by a CNN. Beyond detecting the presence of fish, studies have shown that deep learning methods can classify

zebrafish individuals [20], even when their appearance changes over time [31], [32]. Generally, *idtracker.ai* presents a versatile tracker which has been demonstrated, e.g., for tracking mice [21] or zebrafish [21]. To train the CNN, so called tracklets, i.e., short term trajectories, are generated ad-hoc. However, there are limitations of this approach, including occlusions of fish limiting the length of tracklets and hence reducing the amount of available training data [26]. Also, optimal parameter settings, such as the blob size, may change during tracking, e.g., if the distance between the camera and the fish changes. Overall, the algorithm performs robustly for land animals [21] or in shallow water tanks by limiting occlusions and restricting changes in image features, e.g., the size of the fish body in pixels. Following, the authors of *idtracker.ai* present an experimental setup with a water depth of 25 mm [33]. Particularly to track zebrafish, the movement along the depth axis of the camera can be limited by divider plates, e.g., to 35 mm [34] or by shallow water depths of 30 mm to 100 mm [35]. This strongly limits the natural movement of the fish, which is crucial for musculoskeletal motion studies.

In this study, we introduce an end-to-end fish-tracking algorithm based on deep learning. Our approach is robust towards low image resolutions, low frame rates, and tracking in water tanks of up to 180 mm in depth. The experimental setup consists of a water tank and two RGB-cameras imaging fish movements from the top of the water tank. We decompose our tracking approach into four stages depicted in Fig. 1: (1) we localize fish with a neural network and crop patches surrounding the fish head, (2) we identify fish individuals based on cropped patches with a second neural network, (3) we create 2D tracklets, and (4) we estimate 3D trajectories. Key contributions include

(1) a simplified approach to acquire and extend the training data, circumventing tedious manual annotation, (2) a tracking approach robust with respect to low image resolutions, and (3) a public dataset with 17.000 annotated labels of up to ten zebrafish swimming freely in the tank.

We systematically evaluate the performance of our tracking approach on videos featuring multiple fish (2, 5, and 10 fish), different patch sizes, camera resolutions, and training data augmentation. Furthermore, we evaluate the efficacy of our 3D tracking algorithm by tracking up to 10 zebrafish over a 30-minute time span and consistently outperform the state-of-the-art tracking approach. Finally, we present activity profiles for individual fish, including velocity, distance traveled, turning angles, and occupancy maps.

II. METHODS

Our general data processing pipeline is shown in Fig. 1. We first feed the acquired image data into a localization model, which estimates bounding boxes (shown in blue) containing the fish heads. Next, we feed image patches $\tilde{x}$ containing a fish head at the center into a classification model, which identifies the individual fish and labels the patches accordingly. Using a sequence of labeled fish head positions, we create 2D tracklets, i.e., sequences of positions for each fish and each camera, separately. Knowing the position and identity of each fish and the extrinsic and intrinsic camera parameters [36], we reconstruct the 3D trajectory of each fish.

A. LOCALIZATION AND CLASSIFICATION

Let D^f be a dataset containing only images showing one individual fish f swimming in the tank. We consider f representing the fish class, i.e., $f \in F = \{1, 2, \dots, k\}$. As a baseline CNN model we use the YOLOv12 framework [37], [38].

1) FISH LOCALIZATION

To generate target labels for training object localization, we apply a blob detection algorithm [39]. These algorithms are robust when there are no occlusions in the images [20], which holds for the videos of fish swimming individually. When viewed from above, a fish's head is partially elliptical and the white background increases the contrast of the fish, resulting in a simple to detect blob. Note, each fish is placed in the tank individually to acquire training data, i.e., only one object is in the images. For each image in D^f we estimate the fish position and use it as the label for training the object localization network. We leverage the training data with image augmentation methods, e.g., rotations ($\pm 10^\circ$), horizontal and vertical flips, mixup, and mosaic with a 0.5 probability. After random shuffling, the dataset was partitioned into 80:10:10 ratios for training, validation, and testing, respectively.

2) FISH CLASSIFICATION

Based on the fish positions, we crop the images in D^f to obtain patches $\tilde{x}$ containing the fish head located at the center,

compare Fig. 1, bottom. As fish swim individually during data acquisition, we readily know the fish class which we use as the label for training a classification network. Each patch $\tilde{x}$ is rotated incrementally to enable robust classification of fish swimming in various directions. The following steps are applied to represent all orientations uniformly in the training data set. First, we estimate the fish head orientation α in a crop $\tilde{x}$ through binarization of the image using Otsu's thresholding algorithm [40] and a principal component analysis. Due to the shape of the fish head, the primary component typically points in the direction of the fish head. Second, we rotate the full resolution $4096 \text{ px} \times 3000 \text{ px}$ image by $-\alpha$. Third, we rotate the image by $\theta = 5^\circ$ counterclockwise for successive frames in D , i.e., frame 1 rotated by 0° , frame 2 rotated by θ , frame 3 by 2θ and so on. Finally, we extract a patch $\tilde{x}$ of size $h \times w$ around the transformed head position, i.e., such that the head is again centered with respect to the patch. We train a robust classifier to handle all fish orientations. In contrast, previous approaches rotate cropped patches of the fish to a specific orientation before classification [20]. We train individual classification networks on patches $\tilde{x}$ of different sizes, i.e., $h \times w$ defined as $40 \text{ px} \times 40 \text{ px}$, $80 \text{ px} \times 80 \text{ px}$, $120 \text{ px} \times 120 \text{ px}$, $160 \text{ px} \times 160 \text{ px}$, $200 \text{ px} \times 200 \text{ px}$, $240 \text{ px} \times 240 \text{ px}$, $280 \text{ px} \times 280 \text{ px}$, and $320 \text{ px} \times 320 \text{ px}$, respectively. Sample images are depicted in Fig. 1. Further, we present an ensemble model based on majority voting of the output of the classification networks with patch sizes $120 \text{ px} \times 120 \text{ px}$, $160 \text{ px} \times 160 \text{ px}$, $200 \text{ px} \times 200 \text{ px}$, and $240 \text{ px} \times 240 \text{ px}$.

B. TRAJECTORY FORMATION

We define a tracklet as a segment of a trajectory that represents the path of a fish over a specific period. A tracklet $T_f^{t \rightarrow t+n}$, where f is the fish class, can be represented as an ordered set of n detections $\{\partial(t), \partial(t+1), \dots, \partial(t+n)\}$ in image frames t . Each detection is formally described as $\partial(t) = (x_t, y_t, c_t, f_t)$, where x_t and y_t denote fish head coordinates in pixels, c_t the prediction confidence score estimated by the model and f_t the fish class. Multiple fish are located inside an image frame leading to N detections defined as $M_f(t) = \{\partial_N(t)\}$. In the most common case, a single detection $\partial(t)$ is given in $M_f(t)$ for a fish class f and $\partial(t)$ is added to the tracklet, as depicted in Fig. 2a. To limit false associations during tracklet generation, we define a confidence threshold $\hat{c}$ and distance threshold $\hat{d}$. Consider multiple detections for a fish class f in image frame $t+1$. Detections $\partial_f(t+1)$ below a confidence threshold $\hat{c}$ are rejected, as depicted in Fig. 2b. Detections exceeding the distance threshold $\hat{d}$ are rejected based on the Euclidean distance between $\partial_f(t)$ and $\partial_f(t+1)$, as depicted in Fig. 2c. These steps ensure the integrity of tracking, encompassing solely high-confidence detections from the same fish while within a reasonable distance. Further, we consider two exceptions: (1) Consider multiple detections that fulfill the distance threshold but have an identical confidence score.

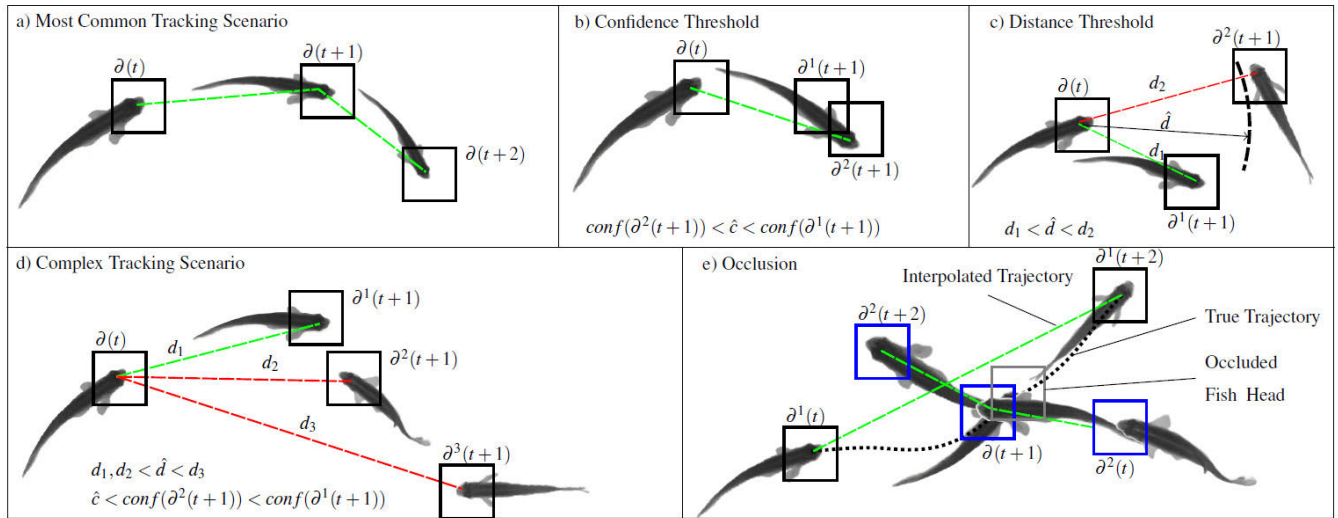


FIGURE 2. Tracklet formation: The Color of the bounding boxes represents the fish class, t denotes the frame number, detection objects are ∂ and the corresponding confidence is given by $\text{conf}(\partial)$, green dashed lines indicate chosen trajectories by our algorithm while red dashed lines indicate rejected trajectories. (a) The most common scenario in which a single fish is accurately detected and classified in consecutive frames. (b) Bounding boxes with a confidence smaller than $\hat{c}$ are rejected. (c) Bounding boxes with a distance larger than $\hat{d}$ are rejected. (d) Configuration where two bounding boxes meet the distance and confidence thresholds. In this unique case, the bounding box with the highest confidence score is chosen. (e) Missing detections (gray bounding box) are linearly interpolated between frames.

In this very unlikely case the detection object ∂_f with the lowest Euclidean distance is chosen. (2) Consider the case where no detection objects $\partial_f(t+1)$ are present for a fish f in subsequent frames, e.g., during an occlusion of fish as depicted in Fig. 2e. In this case, a new tracklet $\mathcal{T}_f$ is initiated and we apply a linear interpolation between tracklets of a fish class f .

C. EVALUATION METRICS

Let k be the total number of fish and T the total number of frames. All metrics are estimated with an intersection over union (IoU) of 0.5.

- 1) The **Multiple Object Tracking Accuracy (MOTA)** is composed of three error ratios:

$$(I) \quad \hat{m} = \frac{\sum_t m_t}{\sum_t g_t}, \quad (II) \quad FP = \frac{\sum_t FP_t}{\sum_t g_t}$$

$$\text{and } (III) \quad mme = \frac{\sum_t mme_t}{\sum_t g_t} \quad (1)$$

with m_t the number of misses in all time t , g_t the total number of objects present in time t , FP_t the number of false positives in time t , and mme_t the total number of incorrectly classified detections in time t . The MOTA is then defined as given in [41] as

$$MOTA = 1 - E \quad \text{with} \quad E = \frac{\sum_t (m_t + FP_t + mme_t)}{\sum_t g_t} \quad (2)$$

Overall, this metric intuitively quantifies the tracker's ability to maintain accurate trajectories, irrespective of its precision in estimating object positions. MOTA

is considered a benchmark metric for multiple object trackers [42].

- 2) The **Multiple Object Tracking Precision (MOTP)** quantifies the localization precision of correctly matched fish. It is estimated by the intersection-over-union (IoU) between the predicted and annotated 50px × 50px bounding boxes and averaged by the number of matches c_t found at time t :

$$MOTP = \frac{\sum_{i,t} (1 - \text{IoU}_t^i)}{\sum_t c_t}, \quad (3)$$

The MOTP is dimensionless and ranges from 0 to 1. Lower values indicate higher localization precision.

- 3) The **Recall** is defined as the number of predicted fish bounding boxes over the total number of bounding boxes.
- 4) The **Detection Rate** is defined as the fraction of correctly predicted bounding boxes b_i over the total amount of bounding boxes. The predicted bounding box with the highest IoU is chosen if multiple bounding boxes overlap with a ground truth bounding box. Formally,

$$\text{DetectionRate} = \frac{\sum_{i=1}^T b_i}{k \cdot T}. \quad (4)$$

- 5) The **Classification Rate** is defined as the number of correctly classified detections and is defined as

$$\text{ClassificationRate} = \frac{\sum_{i=1}^T c_i}{k \cdot T} \quad (5)$$

where c_i is the total number of correct classifications.

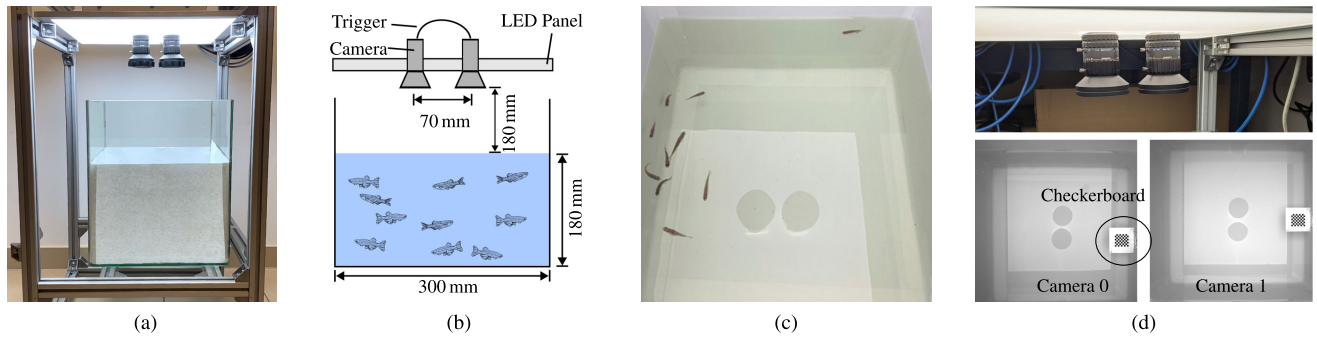


FIGURE 3. Experimental setup: (a-b) Our experimental setup for data acquisition consists of a water tank, two synchronized 4k cameras mounted on the top, and an LED panel for homogeneous illumination. (c) Reflections in the water are minimized by a foil attached to the inside of the water tank. (d) A checkerboard is placed at various positions inside the water tank to estimate the extrinsic and intrinsic parameters of the cameras.

TABLE 1. Image data complexity: The fish head positions are annotated in each frame by hand. We define this data as our ground truth data set.

Number of Fish in Tank	Number of Images	Occlusions		Number of Occluded Patches $\bar{x}$
		Occurrences	Mean Duration	
2 Fish	1000	0	-	0
5 Fish	1000	8	171.5 ± 135.0 ms	26
10 Fish	1000	77	133.5 ± 103.0 ms	212

- 6) The **Identification F1 Score (IDF1)** estimates the identity preservation and how well a tracker maintains the correct identity of each fish over time [43].

We further estimate activity metrics of individual fish, particularly distance traveled, velocity, acceleration, occupancy maps, and turning angles. Estimates are performed between two subsequent detections. We report 2D and 3D tracking metrics for MOTA, MOTP, and Recall. A bounding box size of $120 \text{ px} \times 120 \text{ px}$ and $20 \text{ mm} \times 20 \text{ mm} \times 20 \text{ mm}$ is defined for 2D and 3D tracking, respectively.

III. MATERIALS

A. EXPERIMENTAL SETUP AND DATA ACQUISITION

Our experimental setup is depicted in Fig. 3. The fish are housed in a $300 \text{ mm} \times 300 \text{ mm} \times 300 \text{ mm}$ tank with a 180 mm water height. The tank interior was lined with non-reflective foil to minimize internal reflections. An LED panel was positioned above the tank in which two cameras (acA4112-20um, Basler, Germany) with a fixed focal lens of 8.5 mm (C11-0824-12M, Basler, Germany) were mounted in parallel 70 mm apart. Both cameras were synchronized with a trigger signal to capture frames simultaneously. We record grayscale images with a spatial resolution of $4096 \text{ px} \times 3000 \text{ px}$ and a temporal resolution of 20 Hz. The cameras were stereo calibrated with a checkerboard consisting of $9 \text{ square} \times 8 \text{ square}$ with a square size of 3 mm, as shown in Fig. 3d. We position the checkerboard inside the water tank and acquire 20 image pairs at various locations. We estimate all camera parameters with Zhang's camera calibration algorithm [44], provided by OpenCV (version 4.10.0). We report a mean reprojection error of 0.39 px.

Please note, 3D trajectory reconstruction can be inaccurate due to waves at the water surface caused by fish motion.

B. EXPERIMENTAL EVALUATION AND DATASETS

We generate a training dataset D by recording videos of individual fish placed inside the water tank. For each fish, we record 10 000 images per camera. The fish are kept in the water tank for 30 minutes prior to data acquisition to show natural swimming behaviors. The training dataset D contains data from 10 individual fish.

We evaluate the tracking performance on three individual videos containing 2, 5, or 10 fish swimming in a cohort in the water tank. For quantitative evaluation, we annotate the position of each fish in 1000 images by hand, composing 50 s of video length for each scenario and a total of 17 000 manually annotated bounding boxes. We refer to this dataset as our ground truth data set. In Table 1, we quantify the complexity of our annotated dataset by reporting the occlusion count and duration for 2, 5, and 10 fish videos. Additionally, we record a 30-minute dataset of 10 fish swimming in the tank to demonstrate robust tracking and motion parameters estimated for individual fish.

C. COMPARATIVE STUDY

We estimate performance metrics for our tracking approach with a by-hand annotated ground truth data set and compare tracking performance to the state-of-the-art *idtracker.ai*. Commonly, the parameters for *idtracker.ai* are defined manually. For a fair comparison, we systematically evaluate the parameters of the blob detection to determine the optimal parameters for our video data. We subtract the background and perform all experiments with the *idtracker.ai*

version 5.2.7 using Python 3.11.7 on a NVIDIA GeForce RTX 4090 GPU. All trainings are performed on 30-minute sequences of up to 10 fish swimming in the tank, which exceeds the recommended 20 s video duration by the authors. Since *idtracker.ai* tracks the fish body centroid, while our approach tracks the fish head, we shift our ground truth labels. First, we crop patches with the fish head positioned at the center. Second, we segment the fish body with flood-fill operations and refine the segmentation with an active contour algorithm. Third, we define the centroid of the segmented fish body as our ground truth tracking target. Segmentations that exceed an area threshold are removed, e.g., during occlusions of fish, and missing values are interpolated with a spline function.

D. ANIMALS AND ETHICS STATEMENT

Ten adult zebrafish (wild type, AB strain from the European Zebrafish Resource Center, Karlsruhe, Germany) were raised in the fish facility at the University Medical Center in Hamburg (UKE) and used to perform the tracking experiments. The animals were housed in a water tank according to international housing standards and as previously described [45], [46]. The fish facility is equipped with light sources that mimic natural light and are timed (14 – 10 h light/dark) to promote natural circadian rhythms. The recordings were conducted during the light cycle. The experiment gained prior approval by the institutional review board of the UKE and the ethics committee of the city of Hamburg (No. N111/21), and the methods were performed in accordance with the approved guidelines.

IV. RESULTS

Initially, we study the tracking performance in the context of patch size $\tilde{x}$, distance threshold $\hat{d}$, and confidence threshold $\hat{c}$. We report a MOTA above 0.91 for $\hat{d} > 200$ px for all tracking scenarios, as depicted in Fig. 4a. Similarly, a MOTA above 0.91 is observed with a confidence threshold of 0.6 for classification predictions Fig. 4b. We further study the IoU threshold for estimating metrics and define an IoU of 0.5 for all experiments to allow a fair comparison to *idtracker.ai*, as indicated in Fig. 4c. The tracking accuracy in terms of MOTA and recall for different patch sizes is given in Table 3. Overall, a patch size of 120 px $\times$ 120 px provides the highest MOTA of 0.97 and a recall of 0.99 for the 10 fish tracking scenario. Only minimal deviations in performance are given for the ensemble model.

We estimate the optimal parameters for *idtracker.ai* based on the 10-fish video data set. We define a blob intensity threshold of 45 and a blob size ranging from 1000 to 5000 pixels, based on the average blob size of 2737.12(72859) px, as depicted in Fig. 5a and Fig. 5b, respectively. Further, we study the choice of IoU, as the shifted ground truth labels might deviate from predictions. Minimal deviations in MOTA and Recall are given for an IoU smaller than 0.5, as depicted in Fig. 5c and Fig. 5d, respectively. The performance metrics for our tracking approach in comparison

to *idtracker.ai* are presented in Table 2. We study the tracking performance for *unprocessed* patches, i.e., the natural orientation of the fish is maintained, vs. *pre-processed* patches, i.e., patches are rotated to a defined orientation. Our results indicate a similar tracking performance for 2 and 5 fish. It stands out that the tracking performance is increased for tracking 10 fish with *unprocessed* patches. The MOTA increases by 7.52 % and the recall by 3.36 % for this more complex tracking scenario. A similar trend is observed for 3D tracking. Compared to *idtracker.ai*, we report a higher accuracy for our tracking approach with an increase in MOTA of 10.69 % and an increased recall of 10.01 % for all tracking scenarios. The localization accuracy (MOTP) is advantageous for our method, demonstrating that the network can more accurately detect the center of fish heads in comparison to fish body centroids. For 2 and 5 fish the identification F1 score is similar. However, for 10 fish, we report a higher IDF1 in direct comparison. Also, tracking metrics deviate more between the two camera streams for *idtracker.ai*, compared to our approach, limiting 3D tracking.

We study the robustness in estimating motion activity parameters of individual fish. The deviation between tracked and annotated distances covered by each fish are depicted in the Bland-Altman plot in Fig. 4d. Naturally, the deviation decreases for larger distances covered. At a distance of 2 m we report an average discrepancy of 1.01(95) % between the tracked and annotated distances covered by each fish. Further, we present occupancy maps of each fish derived from tracked and ground-truth data with only minimal deviations (Fig. 4e). Overall, motion parameters can be estimated accurately with our tracking approach. We also evaluate the robustness of our approach for different image resolutions, studying potential low-resolution tracking. As expected, tracking accuracy decreases with lower image resolutions. We report a MOTA of 0.83 for tracking 10-fish with a 50 % reduced image resolution with 1920 px $\times$ 1080 px, as depicted in Fig. 6a. A similar trend is observed for the recall, as shown in Fig. 6b. Further, our approach is robust towards fish head detection in low-resolution images as shown in Fig. 6c. In contrast, the classification rate decreases steadily for lower resolutions, as shown in Fig. 6d. It can be assumed that classifying fish in low-resolution image data is more complex than fish head detection.

Multiple motion parameters are depicted in Fig. 7 for tracking 10-fish over a 30 minute time span. Each fish demonstrates a unique activity profile in terms of distance traveled, velocity, acceleration, aggregated turns, and space occupied in the tank. Fish 8 covers the most distance reaching 202.56 m (Fig. 7a). The uniqueness of fish 8 can also be observed in the occupancy maps (Fig. 7e), as it swims predominantly at the border of the tank. In contrast, fish 7 covers the lowest distance with 90.53 m and predominantly stays at the center of the tank. We estimate a linear correlation model (Fig. 7d) between the overall distance traveled and the body length per fish. No significant correlation is given

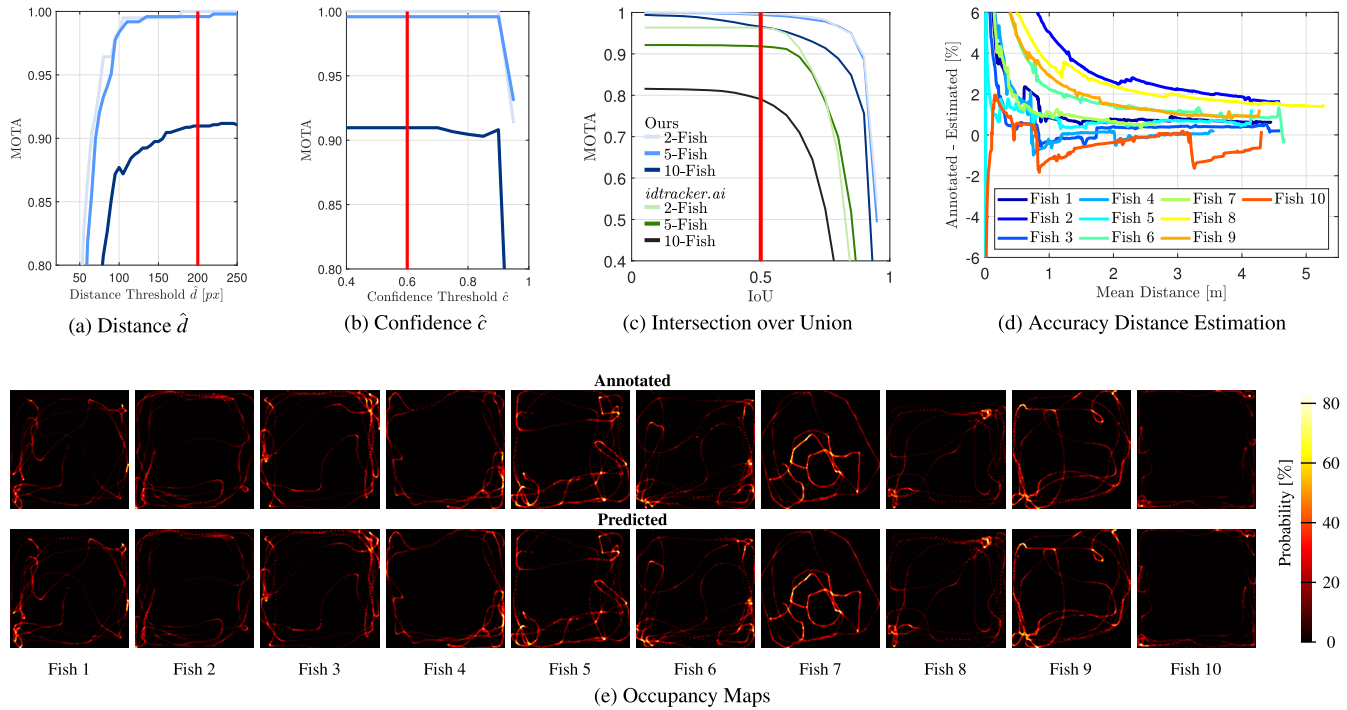


FIGURE 4. Tracking performance metrics All metrics displayed are estimated for tracking on 1000 frames. Ground truth annotations were performed by hand. The red line indicates parameters defined for all experiments performed. Mean 2D MOTA of both camera streams (a) vs. distance threshold and (b) vs. confidence threshold $\hat{c}$. (c) IoU vs. MOTA for our method in comparison to *idtracker.ai*. (d) Bland-Altman plot for distance estimation. (e) Selected occupancy maps.

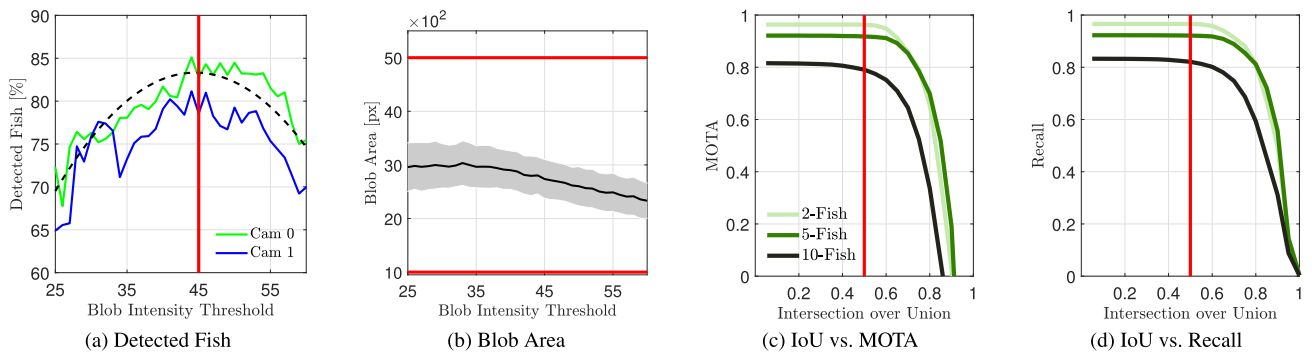


FIGURE 5. Parameter identification *idtracker.ai*: We estimate the optimal parameters for (a) the blob intensity threshold and (b) the blob area size. To allow a fair comparison to our method, we define an IoU of 0.5 for all experiments; IoU vs. (c) MOTA and (d) recall. The red line indicates the set parameters chosen for all experiments performed with *idtracker.ai*.

($P > 0.05$). A positive trend is indicated with an adjusted R^2 value of 0.211.

V. DISCUSSION

We propose a tracking approach that differs from current tracking strategies as we decompose the tracking problem into a localization and classification task while each task is managed by an individual neural network. Advantages of this approach include three main aspects; (1) Fish can be added and removed from the cohort while only the classifier training set needs to be expanded. To this end, we present a straightforward approach for training data acquisition by recording short video sequences of individual

fish swimming in confinement within the tank. We apply conventional image processing methods to localize fish and generate robust training data labels without any tedious manual labeling. Subsequently, a classifier is trained and tailored to identify each specific fish. Please note that the fish localization model remains unchanged. In contrast, several existing tracking methods [20], [21], [47], [48] rely on the ad-hoc learning of classification models. In these studies, a training data set is created by identifying tracklets where fish swim without occlusion. Finding occlusion-free tracklets for the classification model can be particularly challenging when group videos contain numerous fish and consequently many occlusions. This limits their applicability

TABLE 2. Comparative tracking performance metrics: Cropped patches $\tilde{x}$ are either *pre-processed* (aligned to a fixed direction) or *unprocessed* (cropped patches are inferred directly). Results are given for a patch size of 120 px $\times$ 120 px for 2D tracking for camera 0 (C0) and camera 1 (C1) as well as 3D tracking.

Metric	Fish	Pre-Processed			Unprocessed			idtracker.ai	
		2D _{C0}	2D _{C1}	3D	2D _{C0}	2D _{C1}	3D	2D _{C0}	2D _{C1}
MOTA $\uparrow$	2	0.982	0.967	0.892	0.997	0.997	0.973	0.946	0.981
	5	0.981	0.976	0.922	0.993	0.995	0.982	0.924	0.912
	10	0.894	0.902	0.710	0.963	0.968	0.899	0.814	0.767
MOTP $\downarrow$	2	0.066	0.054	0.157	0.048	0.040	0.146	0.116	0.134
	5	0.061	0.052	0.149	0.051	0.041	0.129	0.113	0.110
	10	0.081	0.071	0.196	0.068	0.055	0.172	0.190	0.117
Recall $\uparrow$	2	0.991	0.984	0.946	0.999	0.999	0.987	0.949	0.983
	5	0.991	0.988	0.961	0.997	0.998	0.991	0.930	0.913
	10	0.949	0.953	0.856	0.982	0.984	0.949	0.844	0.798
IDF1 $\uparrow$	2	0.991	0.984	0.946	0.999	0.999	0.981	0.972	0.990
	5	0.991	0.988	0.961	0.997	0.998	0.991	0.940	0.927
	10	0.946	0.950	0.855	0.980	0.983	0.949	0.870	0.839

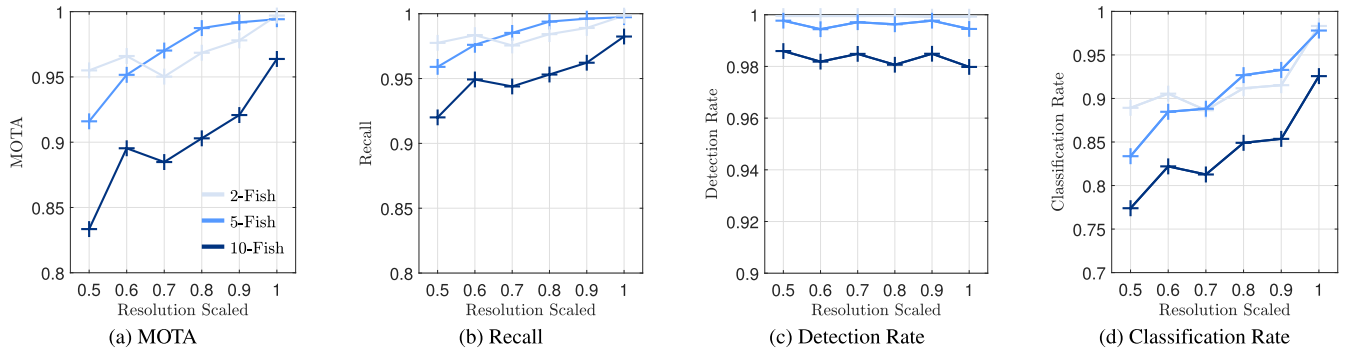


FIGURE 6. Low-resolution tracking performance: We estimate tracking metrics for scaled resolutions of the original 4096 pixels $\times$ 3000 pixels image resolution; (a) MOTA, (b) Recall, (c) Detection Rate, and (d) Classification Rate.

TABLE 3. Tracking performance for different patch sizes: 2D tracking performance metrics for different patch sizes $\tilde{x}$. The arrows indicate which metric values are desirable. Metrics are estimated for tracking 10-fish and patches are *unprocessed* (no pre-processing). Patch sizes with an asterisk (*) are utilized for the ensemble model.

Size Patch $\tilde{x}$	MOTA $\uparrow$			Recall $\uparrow$			IDF1 $\uparrow$		
	Cam 0	Cam 1	Mean	Cam 0	Cam 1	Mean	Cam 0	Cam 1	Mean
40 $\times$ 40 px	0.804	0.839	0.821	0.905	0.923	0.914	0.895	0.916	0.906
80 $\times$ 80 px	0.875	0.897	0.886	0.940	0.951	0.946	0.934	0.946	0.940
120 $\times$ 120 px*	0.963	0.968	0.965	0.982	0.984	0.983	0.980	0.983	0.983
160 $\times$ 160 px	0.917	0.931	0.924	0.960	0.968	0.964	0.956	0.963	0.960
200 $\times$ 200 px	0.916	0.906	0.911	0.960	0.956	0.958	0.957	0.951	0.954
240 $\times$ 240 px	0.893	0.914	0.903	0.950	0.960	0.955	0.942	0.953	0.948
280 $\times$ 280 px	0.846	0.916	0.881	0.928	0.961	0.944	0.919	0.955	0.937
320 $\times$ 320 px	0.823	0.890	0.857	0.916	0.948	0.932	0.906	0.941	0.924
Ensemble*	0.965	0.962	0.963	0.983	0.982	0.982	0.982	0.980	0.981

for complex tracking scenarios. In comparison, our training approach is not limited by the number of fish swimming in the tank. Recently, contrastive learning has been evaluated in *idtracker.ai* to learn meaningful representations that cluster animal identities [49]. (2) Our approach relies on a simple linking algorithm of tracklets due to the robust classifier. The classifier predicts the same fish identity in both camera streams. Hence, we do not rely on complex

matching algorithms between 2D tracklets of the two camera streams, and 3D trajectory formation is streamlined. Recently, general deep learning-based tracking methods have been proposed [50], [51]. However, these networks need to be adapted to tracking zebrafish, as they have a unique motion profile, e.g., abrupt changes in direction and strongly fluctuating velocity. Finally, (3) we present an open-access dataset for RGB stereo camera tracking of zebrafish. Our

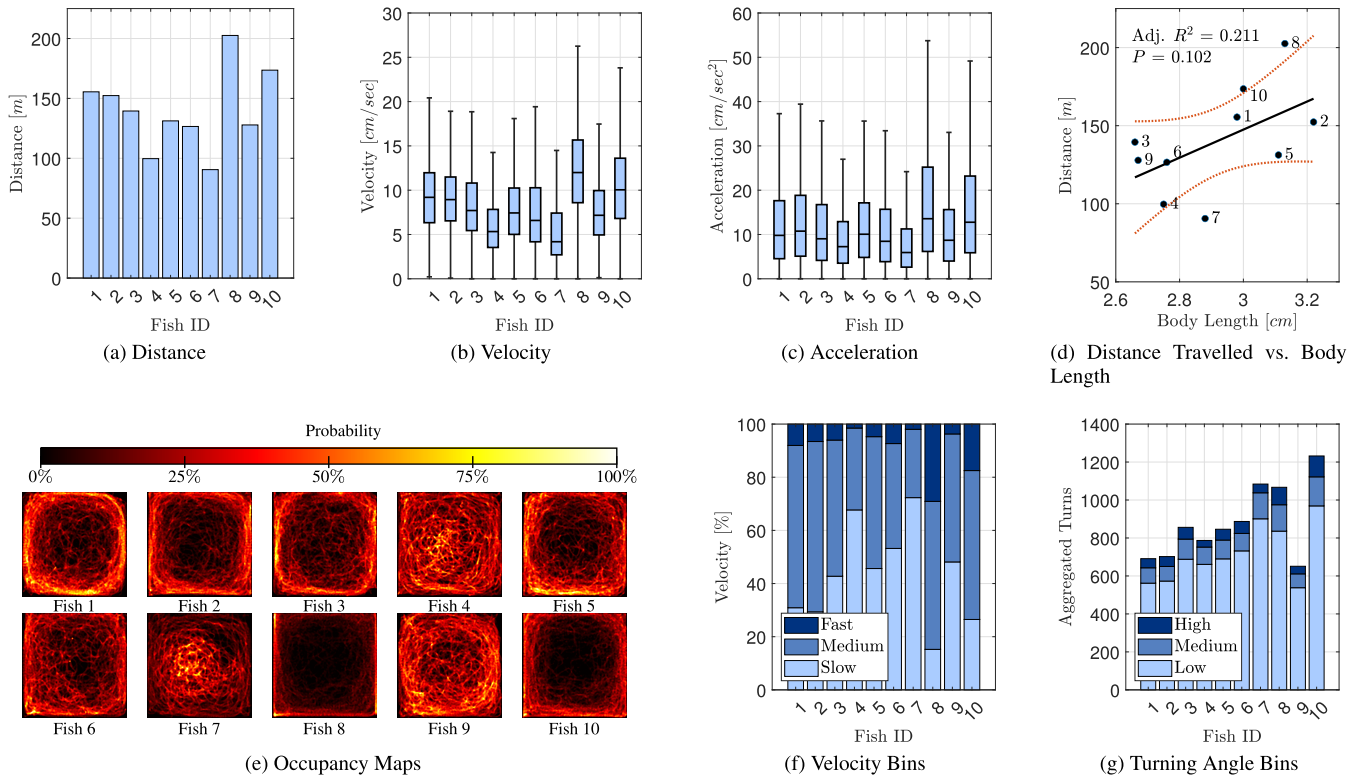


FIGURE 7. Motion profiles All experiments displayed are performed on a 30-minute video sequence. Given are: (a) total distance traveled, (b) velocity (c) acceleration, and (d) distance traveled vs. body length of each fish. The red dashed line indicates a linear regression model's 95% confidence interval. (e) Occupancy map highlighting the areas occupied by each fish in the fish tank. (f) Percentage of slow ($<7 \text{ cm s}^{-1}$), fast ($>15 \text{ cm s}^{-1}$) and medium movements of each individual fish. (g) Aggregated turns of each fish grouped in low ($20^\circ\text{--}40^\circ$), medium ($40^\circ\text{--}60^\circ$) and high ($>60^\circ$).

experimental setup allows tracking of fish movements with two RGB cameras mounted at the top of the tank. Previous open-access datasets recorded images from two different viewpoints and at a lower spatial resolution [52]. Our dataset includes 17,000 annotated labels of up to 10 zebrafish swimming freely in the tank, along with a 30-minute sequence including the predicted labels.

We perform an extensive analysis for our methods and present robust default settings. Fig. 4a and Fig. 4b illustrate that the tracking performance is not particularly sensitive with respect to the distance and confidence thresholds, and values of 200 px and 0.6 worked for all experiments and different numbers of fish. Considering the size of patches used to identify fish, our results indicate the highest tracking performance for a size of $120 \text{ px} \times 120 \text{ px}$. This size typically contains the fish head and partial body while limiting the amount of unrelated background in the patch. Note that the size is related to the camera's resolution and that the fraction of a fish covered by a patch also depends on its distance from the camera. Additionally, the proposed method would robustly work with different image resolutions. The detection performance is virtually independent down to half the resolution of our cameras, with detection rates above 98%. In contrast, the rate of correctly classified fish is reduced from 93% to approximately 77% for tracking with half the camera resolution. This shows that high resolution fish

features of the head, e.g., the shape, are crucial for robust classification.

To illustrate the performance of our approach we directly compare it to *idtracker.ai*, which is considered state-of-the-art. Note, that we carefully tuned the parameters for *idtracker.ai* to obtain a fair comparison, see Fig. 5. Further, we carefully follow the guidelines for sufficient video quality indicated on the *idtracker.ai* homepage. For 2D tracking our proposed methods achieves a mean MOTA of 0.986 in comparison to a mean MOTA of 0.890 for *idtracker.ai*. Moreover, for the most complex case of tracking 10 fish, the tracking performance between the two camera streams only slightly differs by 0.52 percentage points for our approach in contrast to 5.78 percentage points for *idtracker.ai*, compare Table 2. This is crucial, as 3D tracking depends on robust detection in both camera streams. The larger deviation for *idtracker.ai* may be related to its sensitivity with respect to some parameters, e.g., blob detection can fail if fish swim closer to the camera. The authors perform their studies in shallow water depth of 25 mm which limits the number of occlusions and allows consistent image features for the blob detection algorithm [33]. Similar approaches have been applied by installing divider plates limiting the width of the tank to 35 mm [34]. However, this also constraints the fish motion substantially and to study natural swimming behavior it would be desirable to track in deeper water.

Pedersen et al. [52] use a setup containing a side and a top camera for tracking 10 fish for 15 s and report a 3D MOTA of 0.25. On the same dataset Wang et al. [48] developed an ad-hoc tracking method and achieved an improved MOTA of 0.67. With our setup and our method, we obtain a substantially higher 3D MOTA of 0.90 for tracking 10 fish in a water depth of 180 mm. Still, it should be noted that the experimental setups differ.

Further, we evaluate the amount of image pre-processing required for successful tracking. Previous approaches proposed a number of pre-processing steps, e.g., rotating patches such that the head points to the same standardized direction [20], [47]. In contrast, we propose to train the classifier on various image patches containing fish swimming in multiple directions. This allows cropping patches from a video stream containing multiple fishes and performing inference on these *unprocessed* patches directly. Table 1 shows only minimal deviations between *unprocessed* and *processed* patches for 2 and 5 fish tracking. However, tracking metrics increase by 7.52 percentage points for 10-fish tracking. Similar trends are also given for MOTP and recall. Hence, more robust estimates are feasible with *unprocessed* patches and fewer pre-processing steps are advantageous.

So far, zebra fish tracking has often been evaluated on shorter sequences, e.g., 15 s [52]. However, one of the interesting applications of tracking fish are long term studies of fish activity. Considering 17.000 manually annotated bounding boxes, we estimate only minimal errors for distance and precise occupancy maps, see Figures 4a and 4e. Note, that these errors are small compared to the expected uncertainty in the manual annotation.

Considering generalization, we expect the method to work with similar setups, i.e., with cameras from top and sufficient contrast between fish and background. Note, that we decouple the trajectory generation, which will only require conventional stereo camera calibration. As illustrated by our analysis of the preferred patch size, the size of the tracking setup may require adjustment depending on the camera resolution and distance, e.g., higher resolution cameras closer to the fish may require larger patches to cover the fish sufficiently. Note, that we provide details of our setup, i.e., for different setups it should be rather simple to estimate suitable camera resolution and patch size settings from our results. Also, the number of cameras can be increased to track even larger volumes or different animal species.

Fig. 7e illustrates that different motion patterns exist. In fact, our results for a 30-minute tracking sequence with 10 fish show distinct activity profiles for individual fish, although all fish were raised from the same batch. For example, fish 7 and 8 exhibit different swimming behavior, the former swimming slow and remaining mostly at the center of the tank, while the latter covers twice the distance by swimming fast along the tank's perimeter. While the correlation between body length and distance is weak and not significant, it illustrates that musculoskeletal features

and motion patterns may be related and need to be further evaluated. In this context, the flexibility of our approach provides a number of advantages, e.g., as fish can be removed or added to the cohort, or fish could be temporarily removed from the tank. Hence, the tracking performance and the ability to effectively study motion patterns underline the usefulness of the approach.

VI. CONCLUSION

In this paper, we introduce a deep learning-based 3D tracking method of zebrafish by employing two CNNs, one for detection and the other for classification. Our approach utilizes an experimental setup with two top-view cameras to establish 2D trajectories, simplifying the subsequent creation of 3D trajectories. Our training data acquisition approach involves capturing videos of individual fish. Despite training exclusively on single fish videos, our network exhibits robustness when applied to scenarios involving multiple fish swimming in a tank, eliminating the need for re-training with images from group videos. Our unique setup further enables the addition of new fish to the cohort if required by simply extending the training data set of the classification network. We achieve a MOTA score of 0.97 in a 10-fish tracking scenario, surpassing the performance of comparable tracking methods.

DATA AVAILABILITY

The data analyzed in this study, the trained models, and a video demonstrating the tracker's performance are available at <https://hdl.handle.net/11420/63594>

REFERENCES

- [1] S. Siddiqui, H. Siddiqui, E. Riguene, and M. Nomikos, "Zebrafish: A versatile and powerful model for biomedical research," *BioEssays*, vol. 47, no. 12, p. 70080, Dec. 2025.
- [2] K. Dietrich, I. A. Fiedler, A. Kurzyukova, A. C. López-Delgado, L. M. McGowan, K. Geurtzen, C. L. Hammond, B. Busse, and F. Knopf, "Skeletal biology and disease modeling in zebrafish," *J. Bone Mineral Res.*, vol. 36, no. 3, pp. 436–458, Dec. 2020.
- [3] L. Lleras-Forero, C. Winkler, and S. Schulte-Merker, "Zebrafish and medaka as models for biomedical research of bone diseases," *Develop. Biol.*, vol. 457, no. 2, pp. 191–205, Jan. 2020.
- [4] D. J. M. Bergen, E. Kague, and C. L. Hammond, (2019) "Zebrafish as an emerging model for osteoporosis: A primary testing platform for screening new osteo-active compounds" *Front. Endocrinol.*, vol. 6, no. 6, doi: [10.3389/fendo.2019.00006](https://doi.org/10.3389/fendo.2019.00006).
- [5] F. Tonelli, J. W. Bek, R. Besio, A. De Clercq, L. Leoni, P. Salmon, P. J. Coucke, A. Willaert, and A. Forlino, "Zebrafish: A resourceful vertebrate model to investigate skeletal disorders," *Frontiers Endocrinology*, vol. 11, p. 489, Jul. 2020.
- [6] R. Y. Kwon, C. J. Watson, and D. Karasik, "Using zebrafish to study skeletal genomics," *Bone*, vol. 126, pp. 37–50, Sep. 2019.
- [7] M. Hur, C. A. Gistelincq, P. Huber, J. Lee, M. H. Thompson, A. T. Monstad-Rios, C. J. Watson, S. K. McMennamin, A. Willaert, D. M. Parichy, P. Coucke, and R. Y. Kwon, "MicroCT-based phenomics in the zebrafish skeleton reveals virtues of deep phenotyping in a distributed organ system," *eLife*, vol. 6, p. 26014, Sep. 2017, doi: [10.7554/elife.26014](https://doi.org/10.7554/elife.26014).
- [8] I. A. K. Fiedler, F. N. Schmidt, E. M. Wölfel, C. Plumeyer, P. Milovanovic, R. Gioia, F. Tonelli, H. A. Bale, K. Jähn, R. Besio, A. Forlino, and B. Busse, "Severely impaired bone material quality in chihuahua zebrafish resembles classical dominant human osteogenesis imperfecta," *J. Bone Mineral Res.*, vol. 33, no. 8, pp. 1489–1499, Aug. 2018.

- [9] J. W. Bek, A. De Clercq, P. J. Coucke, and A. Willaert, "The ZE-tunnel: An affordable, easy-to-assemble, and user-friendly benchtop zebrafish swim tunnel," *Zebrafish*, vol. 18, no. 1, pp. 29–41, Feb. 2021.
- [10] C. J. Watson et al., "Wnt16 regulates spine and muscle morphogenesis through parallel signals from notochord and dermomyotome," *PLOS Genet.*, vol. 18, no. 11, Nov. 2022, Art. no. e1010496.
- [11] Y. Yang, F. Turci, E. Kague, C. L. Hammond, J. Russo, and C. P. Royall, "Dominating lengthscales of zebrafish collective behaviour," *PLOS Comput. Biol.*, vol. 18, no. 1, pp. 1–14, Jan. 2022, doi: [10.1371/journal.pcbi.1009394](https://doi.org/10.1371/journal.pcbi.1009394).
- [12] S. Shreesha, M. M. M. Pai, U. Verma, and R. M. Pai, "Fish tracking and continual behavioral pattern clustering using novel sillago sihama vid (SSVid)," *IEEE Access*, vol. 11, pp. 29400–29416, 2023.
- [13] X. Liu, Y. Yue, M. Shi, and Z.-M. Qian, "3-D video tracking of multiple fish in a water tank," *IEEE Access*, vol. 7, pp. 145049–145059, 2019.
- [14] Z.-M. Qian, S. H. Wang, X. E. Cheng, and Y. Q. Chen, "An effective and robust method for tracking multiple fish in video image based on fish head detection," *BMC Bioinf.*, vol. 17, no. 1, pp. 1–11, Jun. 2016, doi: [10.1186/s12859-016-1138-y](https://doi.org/10.1186/s12859-016-1138-y).
- [15] Y.-X. Bai, S.-H. Zhang, Z. Fan, X.-Y. Liu, X. Zhao, X.-Z. Feng, and M.-Z. Sun, "Automatic multiple zebrafish tracking based on improved HOG features," *Sci. Rep.*, vol. 8, no. 1, pp. 1–14, Jul. 2018, doi: [10.1038/s41598-018-29185-0](https://doi.org/10.1038/s41598-018-29185-0).
- [16] M. M. Saberioon and P. Cisar, "Automated multiple fish tracking in three-dimension using a structured light sensor," *Comput. Electron. Agricult.*, vol. 121, pp. 215–221, Feb. 2016, doi: [10.1016/j.compag.2015.12.014](https://doi.org/10.1016/j.compag.2015.12.014).
- [17] X. Zhao, S. Yan, and Q. Gao, "An algorithm for tracking multiple fish based on biological water quality monitoring," *IEEE Access*, vol. 7, pp. 15018–15026, 2019.
- [18] S. H. Wang, J. W. Zhao, and Y. Q. Chen, "Robust tracking of fish schools using CNN for head identification," *Multimedia Tools Appl.*, vol. 76, no. 22, pp. 23679–23697, Nov. 2017.
- [19] Y.-L. Fan, C.-H. Hsu, F.-R. Hsu, and L.-D. Liao, "Exploring the use of deep learning models for accurate tracking of 3D zebrafish trajectories," *Frontiers Bioeng. Biotechnol.*, vol. 12, May 2024, Art. no. 1461264.
- [20] Z. Xu and X. E. Cheng, "Zebrafish tracking using convolutional neural networks," *Sci. Rep.*, vol. 7, pp. 1–11, Feb. 2017.
- [21] F. Romero-Ferrero, M. G. Bergomi, R. C. Hinz, F. J. H. Heras, and G. G. de Polavieja, "Idtracker.AI: Tracking all individuals in small or large collectives of unmarked animals," *Nature Methods*, vol. 16, no. 2, pp. 179–182, Feb. 2019.
- [22] M. D. O. Barreiros, D. D. O. Dantas, L. C. D. O. Silva, S. Ribeiro, and A. K. Barros, "Zebrafish tracking using YOLOv2 and Kalman filter," *Sci. Rep.*, vol. 11, no. 1, pp. 1–14, Feb. 2021, doi: [10.1038/s41598-021-81997-9](https://doi.org/10.1038/s41598-021-81997-9).
- [23] Y. Xu, Y. Jin, Y. Zhang, Q. Zhu, Y. He, and H. Sheng, "3D zebrafish tracking with topology association," *IET Image Process.*, vol. 17, pp. 1044–1059, Mar. 2023.
- [24] W. Xu, Z. Zhu, F. Ge, Z. Han, and J. Li, "Analysis of behavior trajectory based on deep learning in ammonia environment for fish," *Sensors*, vol. 20, no. 16, p. 4425, Aug. 2020.
- [25] K. Banno, H. Kaland, A. M. Crescitelli, S. A. Tuene, G. H. Aas, and L. C. Gansel, "A novel approach for wild fish monitoring at aquaculture sites: Wild fish presence analysis using computer vision," *Aquaculture Environ. Interact.*, vol. 14, pp. 97–112, Aug. 2022.
- [26] Z. Wang, C. Xia, and J. Lee, "Parallel fish school tracking based on multiple appearance feature detection," *Sensors*, vol. 21, no. 10, p. 3476, May 2021.
- [27] A. Jalal, A. Salman, A. Mian, M. Shortis, and F. Shafait, "Fish detection and species classification in underwater environments using deep learning with temporal information," *Ecological Informat.*, vol. 57, Apr. 2020, Art. no. 101088.
- [28] M. Elmezain, L. S. Saoud, A. Sultan, M. Heshmat, L. Seneviratne, and I. Hussain, "Advancing underwater vision: A survey of deep learning models for underwater object recognition and tracking," *IEEE Access*, vol. 13, pp. 17830–17867, 2025.
- [29] W. Li, Y. Liu, W. Wang, Z. Li, and J. Yue, "TFMFT: Transformer-based multiple fish tracking," *Comput. Electron. Agricult.*, vol. 217, Feb. 2024, Art. no. 108600.
- [30] Y. Liu, B. Li, X. Zhou, D. Li, and Q. Duan, "Fishtrack: Multi-object tracking method for fish using spatiotemporal information fusion," *Expert Syst. Appl.*, vol. 238, May 2024, Art. no. 122194.
- [31] J. Puchalla, A. Serianni, and B. Deng, "Zebrafish identification with deep CNN and ViT architectures using a rolling training window," *Sci. Rep.*, vol. 15, no. 1, pp. 1–13, Mar. 2025.
- [32] D. Cao, C. Guo, Y. Cheng, W. Zhang, M. Shi, and X.-Q. Xia, "Longitudinal identification of zebrafish individuals by deep learning," *BioRxiv*, vol. 2025, pp. 1–12, May 2025.
- [33] A. Pérez-Escudero, J. Vicente-Page, R. C. Hinz, S. Arganda, and G. G. De Polavieja, "IdTracker: Tracking individuals in a group by automatic identification of unmarked animals," *Nature Methods*, vol. 11, no. 7, pp. 743–748, Jul. 2014.
- [34] J. B. Haurum, A. Karpova, M. Pedersen, S. H. Bengtson, and T. B. Moeslund, "Re-identification of zebrafish using metric learning," in *Proc. IEEE Winter Appl. Comput. Vis. Workshops (WACVW)*, Mar. 2020, pp. 1–11.
- [35] S. H. Wang, X. E. Cheng, Z.-M. Qian, Y. Liu, and Y. Q. Chen, "Automated planar tracking the waving bodies of multiple zebrafish swimming in shallow water," *PLoS ONE*, vol. 11, no. 4, Apr. 2016, Art. no. e0154714.
- [36] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
- [37] Y. Tian, Q. Ye, and D. Doermann, "YOLOv12: Attention-centric real-time object detectors," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 38, 2025, pp. 87151–87175. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2025/file/7103444259031cc58051f8c9a4868533-Paper-Conference.pdf
- [38] Y. Tian, Q. Ye, and D. Doermann. (2025). *YOLO12: Attention-Centric Real-Time Object Detectors*. [Online]. Available: <https://github.com/sunsmarterjie/yolov12>
- [39] T. Lindeberg, "Scale selection properties of generalized scale-space interest point detectors," *J. Math. Imag. Vis.*, vol. 46, no. 2, pp. 177–210, Jun. 2013.
- [40] N. Otsu, P. L. Smith, D. B. Reid, C. Environment, L. Palo, P. Alto, and P. L. Smith, "Otsu 1979 Otsu method," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.
- [41] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *EURASIP J. Image Video Process.*, vol. 2008, no. 1, pp. 1–10, 2008.
- [42] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "MOT16: A benchmark for multi-object tracking," 2016, *arXiv:1603.00831*.
- [43] E. Ristani, F. Solera, R. S. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 17–35.
- [44] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.
- [45] P. Aleström, L. D'Angelo, P. J. Midtlyng, D. F. Schorderet, S. Schulte-Merker, F. Sohm, and S. Warner, "Zebrafish: Housing and husbandry recommendations," *Lab. Animals*, vol. 54, no. 3, pp. 213–224, Sep. 2019.
- [46] S. Suniaga, T. Rolvien, A. Vom Scheidt, I. A. K. Fiedler, H. A. Bale, A. Huysseune, P. E. Witten, M. Amling, and B. Busse, "Increased mechanical loading through controlled swimming exercise induces bone formation and mineralization in adult zebrafish," *Sci. Rep.*, vol. 8, no. 1, p. 3646, Feb. 2018.
- [47] X. E. Cheng, S. S. Du, H. Y. Li, J. F. Hu, and M. L. Chen, "Obtaining three-dimensional trajectory of multiple fish in water tank via video tracking," *Multimedia Tools Appl.*, vol. 77, no. 18, pp. 24499–24519, Sep. 2018.
- [48] C. Wang, Z. Wu, Y. Chen, W. Zhang, W. Ke, and Z. Xiong, "Improving 3-D zebrafish tracking with multiview data fusion and global association," *IEEE Sensors J.*, vol. 23, no. 15, pp. 17245–17259, Aug. 2023.
- [49] J. Torrents, T. Costa, and G. de Polavieja, "New idTracker. AI rethinks multi-animal tracking as a representation learning problem to increase accuracy and reduce tracking time," *Elife*, vol. 14, May 2026, Art. no. 107602.
- [50] C. Doersch, A. Gupta, L. Markeeva, A. Recasens, L. Smaira, Y. Aytar, J. Carreira, A. Zisserman, and Y. Yang, "TAP-vid: A benchmark for tracking any point in a video," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 13610–13626.
- [51] N. Karaev, I. Rocco, B. Graham, N. Neverova, A. Vedaldi, and C. Rupprecht, "CoTracker: It is better to track together," in *Proc. Eur. Conf. Comput. Vis.*, 2025, pp. 18–35.
- [52] M. Pedersen, J. B. Haurum, S. Hein Bengtson, and T. B. Moeslund, "3D-ZeF: A 3D zebrafish tracking benchmark dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2423–2433.



imaging modalities (OCT or Ultrasound) are of his research interest.

MAXIMILIAN NEIDHARDT received the M.S. degree in medical engineering from the Technical University of Hamburg, Germany, where he is currently pursuing the Ph.D. degree with the Institute of Medical Technology and Intelligent Systems. His main research focus lies in medical image data processing with deep learning, robotics, and VR applications. Additionally, deep learning based tracking approaches of particularly soft-tissue structures with RGB and sub-surface



NICO FISCHER received the M.S. degree in medical engineering from the Technical University of Hamburg, Germany, in 2026. During his studies, he worked as a Research Assistant at the Institute of Medical Technology and Intelligent Systems (TUHH) and the Institute of Osteology and Biomechanics (UKE). His interests include deep learning based tracking methods, multi-scale material characterization with focus on bone mineralization, and deep-learning based diagnostic in medicine.



DEBAYAN BHATTACHARYA received the M.S. degree in information and communication systems and the Ph.D. degree from the Technical University of Hamburg, Germany, in 2021 and 2025, respectively. He is currently leading ML and AI efforts at Airbus. His research interests include supervised and unsupervised learning for detecting tumors in paranasal sinuses.



BJÖRN BUSSE received the degree in bioengineering and the Ph.D. degree (Hons.) from the Free and Humboldt University Berlin/Charité, where he specialized in bone imaging and mineralization using scanning/backscattered electron microscopy, microanalysis, materials testing, and bone histomorphometry. He is currently a Full Professor of biomedical sciences and medical technology at the Department of Osteology and Biomechanics, University Medical Center Hamburg, Germany, and director of the Interdisciplinary Competence Center for Interface Research (ICCIR). His work focuses on skeletal imaging, biomechanics, and bone quality analysis. His research advances understanding of bone fracture mechanisms, hierarchical tissue interfaces, and bone quality, and addresses skeletal aging, osteoporosis and its treatments, osteomalacia, osteogenesis imperfecta, and Paget's disease. He gained additional research and teaching experience at Lawrence Berkeley National Laboratory and UC Berkeley. He has published more than 140 manuscripts. He has received multiple awards including the John Haddad Young Investigator Award and the von Recklinghausen Prize.



functional performance of whole organisms.

IMKE A. K. FIEDLER received the M.S. degree in biomedical engineering from Hamburg University of Technology and the Ph.D. degree in biomedical sciences from University Medical Center Hamburg (UKE), focusing on establishing zebrafish models in the field of skeletal health. As a Scientist at the Interdisciplinary Competence Center for Interface Research (ICCIR) Hamburg, her focus is on multi-scale imaging from the ultrastructural level of musculoskeletal tissues to



SANDIPAN MUKHERJEE received the B.S. degree in mechatronics engineering from the SRM University of Technology, Chennai, and the M.S. degree in mechatronics engineering from TUHH, Hamburg, in 2024. He is currently a Software Engineer at dpv-analytics GmbH. His research interests include tracking of soft tissue and detection of cardiac arrhythmia and anomalies from EKG using statistical methods and artificial intelligence.



ALEXANDER SCHLAEFER (Senior Member, IEEE) is currently a Professor at Hamburg University of Technology, where he heads the Institute of Medical Technology and Intelligent Systems. His research interests include robotics, navigation, and machine learning, particularly for clinical applications. He is also involved at the Artificial Intelligence Competence Centre for Sustainable Living and Working (SustAIInLivWork) and the Interdisciplinary Competence Center for Interface Research (ICCIR).

...