Three-dimensional reconstruction of the fast-start swimming kinematics of densely schooling fish

Sachit Butail[1] and Derek A. Paley[1,2]

[1]Department of Aerospace Engineering, [2]Neuroscience and Cognitive Science Program,
University of Maryland, College Park, MD 20742, USA

***Abstract*—Information transmission via nonverbal cues such as a fright response can be quantified in a fish school by reconstructing individual fish motion in three dimensions. In this paper we describe an automated tracking framework to reconstruct the full-body trajectories of densely schooling fish using two-dimensional silhouettes in multiple cameras. We model the shape of each fish as a series of elliptical cross sections along a flexible midline. We estimate the size of each ellipse using an iterated extended Kalman filter. The shape model is used in a model-based tracking framework in which simulated annealing is applied at each step to estimate the midline. Results are presented for eight fish with occlusions. The tracking system is currently being used to investigate fast-start behavior of schooling fish in response to looming stimuli.**

***Index Terms*—Target tracking, model-based tracking, schooling fish, giant danio**

I. Introduction

Animal aggregations in many species fascinate and inspire engineers who study collective behavior [1], [2]. Engineering tools have the potential to advance the understanding of animal groups, and roboticists can use this improved understanding to design bio-inspired robotic systems. Among the many animals that demonstrate collective behavior, fish are particularly attractive as a model system because a wide variety of schooling fish are easy to procure and maintain in a laboratory setting.

While there are many bio-inspired algorithms that seek to replicate collective behavior [3], [4], [5], we are not aware of any algorithm that has been validated by experimental data. One reason such experiments are lacking is that (markerless) tracking of multiple organisms is inherently hard. The application of computer-vision techniques has helped, but a technique to track the pose (i.e., position and orientation) and shape of individual animals in a group is not yet available. Even in a laboratory setting, we must address challenges such as underwater lighting, occlusions, and reflections.

Our interest in collective behavior lies in the rapid transmission of information via a nonverbal cue such as a fright response. An example of a fright response in fish is a fast start, which is often the precursor to an escape or attack [6]. Two behaviors associated with fast-start swimming are C-starts and S-starts [7], named for the corresponding body shape during the maneuvers, which take place in less than 100 ms. The propagation of startle responses in a fish school may be indicative of the social transmission of information [8].

Fish schools have been tracked in their natural environment [9] and in laboratories [10], [11]. Positions of up to fourteen fish have been tracked in two dimensions [11] and groups of four and eight fish have been tracked in three dimensions [10]. In [9], an acoustic sensor is used on a moving platform to track individual fish in a school. In [10], [11] least-squares fitting is used to join track segments already matched in sequential video images. In each instance, the fish are modeled as point masses; orientation and shape information is ignored. Shape kinematics have been tracked and studied for fewer fish [12], [13] and the midline has been used previously to describe fish movement [12], [13], [14]. For example, in [13], a two-dimensional model built around the midline is used for tracking.

Deformable objects such as a fish body can be detected in images using active contours [15], [16]. A pre-defined contour based on a decreasing energy function is wrapped around the edges of regions of high-contrast. In three dimensions, deformable objects are encountered in markerless human motion capture [17] and articulated hand tracking [18]. Most of these techniques rely on a predefined three-dimensional model to estimate pose and shape from two-dimensional images. Changes in shape are captured by deforming the model along degrees of freedom such as joint angles or principal components. Methods to define a (deformable) shape use quadrics [18], [19], superquadrics [20], and cubic splines [13]. In [21], we propose approximating fish shape by a bendable ellipsoid. We are able to track simple motion using this method, but not C- or S-starts, which motivates the approach described here.

The number of fish or, more importantly, the density of fish poses another challenge to tracking. For example, it is desirable to preserve the identity of each fish through time and between camera views, even during occlusions. Data-association problems such as this can be addressed instantaneously using shape fitting [22] or over a section of the target trajectory using motion coherence [23], [24], [25]. These problems have been addressed in tracking flies [26], [27], [28] and ants [29]. Data association can be resolved using motion coherence if the occlusions last for only a few frames and the target size is relatively small (so that it is rare for a target to change course while occluded). However, in the case of high frame-rate tracking of fast-start behavior, occlusions can last many frames and the fish often turn while occluded.

In this paper we describe a high-frame-rate tracking framework for estimating the instantaneous shape of multiple fish in a dense school (i.e., with sustained occlusions). We apply methods from generative modeling to produce a shape model, which is then used to reconstruct the fish body in three dimensions using two-dimensional silhouettes in multiple cameras. The contributions of the paper are (1) a method to automatically generate a three-dimensional model of a fish from two orthogonal camera views; and (2) the design of a multi-layered tracking system that reconstructs the position, orientation, and shape of individual fish in a dense school. The technical approach involves the application of tools from generative modeling, nonlinear optimization, and Bayesian estimation.

In our tracking framework, we describe each fish by its position, orientation, and shape (midline). The measurements consist of images from multiple cameras that are each modeled as a perspective-projection system. (A perspective projection is a nonlinear mapping between a three-dimensional point in space and its two-dimensional position in the image plane.) In order to capture the C and S shapes associated with fast-start behavior, we model the midline of the fish body as a polynomial curve. We assign an orthogonal reference frame to each point on the midline and use this frame to automatically construct a three-dimensional shape profile for each fish. We use simulated annealing to optimize the instantaneous state estimate and Kalman filtering to smooth the estimate in time.

The paper is organized as follows. In Section II we introduce the concepts of nonlinear estimation, generative modeling, data association, and nonlinear optimization. Section III presents the fish-midline representation and automatic model generation. Section IV describes a multi-layered approach to reconstruct midline trajectories, including the objective function used in optimization. Section V presents tracking results with up to eight giant danio (*Danio aequipinnatus*). We conclude in Section VI with a description of our ongoing use of the tracking system to study information transmission in danio.

II. Background

A. Nonlinear estimation and data association

In the tracking framework described below, we perform estimation in two stages. First, we estimate the shape geometry of each fish, then we use the estimated shape for model-based tracking. The shape-estimation process uses occluding contours (silhouette boundaries) from multiple views. The model-based tracking uses the shape geometry to reconstruct the fish position, orientation, and midline.

In general, the state of a target at time k is described by the vector $\boldsymbol{X}_k \in \mathbb{R}^n$. A measurement at time k is denoted by $\boldsymbol{Z}_k \in \mathbb{R}^m$. The state $\boldsymbol{X}_{k+1}$ and measurements $\boldsymbol{Z}_{k+1}$ are related to the state $\boldsymbol{X}_k$ according to

$$\begin{aligned} \boldsymbol{X}_{k+1} &= \boldsymbol{F}(\boldsymbol{X}_k, \boldsymbol{w}_{k+1}) \\ \boldsymbol{Z}_{k+1} &= \boldsymbol{H}(\boldsymbol{X}_{k+1}, \boldsymbol{n}_{k+1}), \end{aligned} \tag{1}$$

where $\boldsymbol{F}$ represents the (nonlinear) motion model, $\boldsymbol{H}$ represents the (nonlinear) measurement model, and $\boldsymbol{w}$ and $\boldsymbol{n}$ are the instantaneous disturbance- and measurement-noise values. Given the state estimate $\hat{\boldsymbol{X}}_k$, the estimation error $\hat{\boldsymbol{X}}_k - \boldsymbol{X}_k$ is a random quantity due to noise and approximation in $\boldsymbol{F}$ and $\boldsymbol{H}$. The conditional probability of a state estimate $p(\hat{\boldsymbol{X}}_k|\boldsymbol{Z}^k)$ given the measurements up to time k, $\boldsymbol{Z}^k$, is called the posterior probability density function (pdf). An optimal Bayesian solution recursively maximizes the posterior pdf. Common applications use possibly sub-optimal solutions that assume Gaussian noise distribution.

Our first application of nonlinear estimation is to estimate the shape of each fish. We parameterize the body surface in three dimensions using methods from generative modeling to identify the model parameters. Generative modeling provides a framework for reconstructing the shape of asymmetrical objects. A generative model may be produced by rotating and translating an object along a trajectory [30]. Formally, a continuous set of transformations are applied on an object shape (also called the generator) to build a generative model. A curve generator of the form $\gamma(u) : \mathbb{R} \to \mathbb{R}^3$ is transformed through a parameterized transformation, $\delta(\gamma(u), s) : \mathbb{R}^3 \times \mathbb{R} \to \mathbb{R}^3$, to form a shape. For example, a cylinder with radius r is produced by choosing

$$\gamma(u) = \begin{bmatrix} \cos u \\ \sin u \\ 0 \end{bmatrix} \text{ and } \delta(\gamma(u), s) = \begin{bmatrix} r\gamma_1 \\ r\gamma_2 \\ s \end{bmatrix}, \tag{2}$$

where $s \in [0, 1]$ and $u \in [0, 2\pi]$. Similarly a cone is produced by decreasing $r = 1 - s$ linearly along the trajectory.

In a vision-based tracking system, a nonlinear estimator such as the extended Kalman filter (EKF), the unscented Kalman filter, or the particle filter is often used [31]. The EKF updates the target estimate by linearizing the measurement and target state about the current estimate. A single update of the EKF is equivalent to a single step of a Gauss-Newton optimization method [32]. We iterate the following EKF algorithm to estimate the shape model of a fish.

Extended Kalman Filter (EKF) Algorithm

```
Input:       Motion model $\boldsymbol{F}$, measurement model $\boldsymbol{H}$, covariance matrices for measurement noise $R$ and disturbance $Q$
Initialize:  State estimate $\boldsymbol{X}_1^-$ and error covariance matrix $P_1^-$, prior to the first measurement
For each time step $k = 1, 2, \ldots$
1: Compute gain matrix: $W_k = P_k^- H_k^T S_k^{-1}$ where $S_k = H_k P_k^- H_k + R_k$ is the measurement prediction covariance and $H_k = \frac{\partial \boldsymbol{H}}{\partial \boldsymbol{X}}(\hat{\boldsymbol{X}}_k^-)$
2: Update state estimate: $\hat{\boldsymbol{X}}_k = \hat{\boldsymbol{X}}_k^- + W_k(\boldsymbol{Z}_k - \boldsymbol{H}(\hat{\boldsymbol{X}}_k^-, \boldsymbol{n}_k))$
3: Update state covariance: $P_k = (1 - W_k H_k) P_k^-$
4: Predict state prior to next measurement: $\hat{\boldsymbol{X}}_{k+1}^- = \boldsymbol{F}(\hat{\boldsymbol{X}}_k, \boldsymbol{w}_{k+1})$
5: Compute covariance: $P_{k+1}^- = F_k P_k F_k^T + Q_k$, where $F_k = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{X}}(\hat{\boldsymbol{X}}_k)$
```

In an iterated EKF we loop steps (1)–(3) until a threshold is reached on the matrix norm of the state covariance P_k.

A multi-target tracking system requires measurements to be matched to targets, a process called data association. A simple and fast data-association strategy called nearest-neighbor matching [24] assigns a measurement to the closest (projected) estimate on the image plane. We compute a metric for the distance between the measurement and the target as a function of the complete midline. This metric makes a nearest-neighbor association reliable, even when the targets are close to one another.

B. Nonlinear optimization

In a high-frame-rate tracking system, the time difference between successive measurements is small. As a result, tracking primarily entails processing the measurements, and does

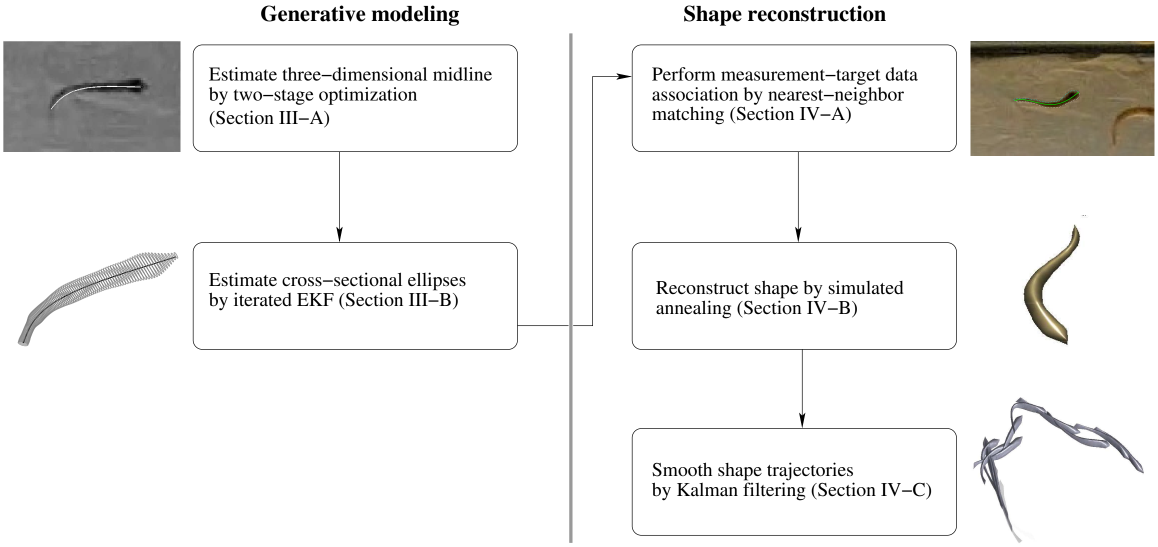

Generative modeling
Shape reconstruction
Estimate three−dimensional midline by two−stage optimization (Section III−A)
Estimate cross−sectional ellipses by iterated EKF (Section III−B)
Perform measurement−target data association by nearest−neighbor matching (Section IV−A)
Reconstruct shape by simulated annealing (Section IV−B)
Smooth shape trajectories by Kalman filtering (Section IV−C)

Fig. 1. Tracking framework. Generative modeling is used to parameterize a shape model; these parameters are estimated using an iterated EKF. Shape reconstruction is performed by matching measurements from segmented images in multiple cameras to a three-dimensional shape estimate.

not require an accurate motion model. For tracking individual fish, we cast the system (1) into a numerical optimization problem and use simulated annealing to solve it. The measurement model is represented by an objective function $\|\boldsymbol{Z}_k - \boldsymbol{H}(\boldsymbol{X}_k, \boldsymbol{n}_k)\|$, which evaluates the match between measurements and the estimate. Simulated annealing (SA) is a probabilistic optimization method used to find the global minimum of the objective function even if there are multiple minima [33]. It mimics the annealing process by accepting a jump out of a local minimum with a probability that decreases as the search approaches a global minimum. The simulated annealing algorithm is summarized in the following table.

Simulated Annealing Algorithm

```
Input:      Cost function $C : \mathbb{R}^n \to \mathbb{R}$, perturbation function $\boldsymbol{r} :$
            $\mathbb{R}^n \to \mathbb{R}^n$, and a non-increasing cooling schedule
Initialize: State estimate at current time-step, $\boldsymbol{X}^1 = \boldsymbol{X}_k$
Until a termination criteria is reached, iterate for $j =$
$1, 2, \ldots$
1: Perturb the system $\tilde{\boldsymbol{X}}^j = \boldsymbol{r}(\boldsymbol{X}^j)$ and compute the costs $C(\boldsymbol{X}^j)$ and
   $C(\tilde{\boldsymbol{X}}^j)$. Let $\delta C$ be the change in cost.
2: Sample from a uniform distribution $\rho \sim \mathbb{U}(0, 1)$ and update the state:
   $\boldsymbol{X}^{j+1} = \begin{cases} \tilde{\boldsymbol{X}}^j & \text{if } \rho \le \min(1, \exp(-\delta C/\tau^j)) \\ \boldsymbol{X}^j & \text{otherwise,} \end{cases}$
   where $\tau^j$ is the temperature.
3: Update the temperature $\tau^j$ based on the cooling schedule (for example,
   $\tau^{j+1} = K_c \tau^j$, where $0 < K_c < 1$.)
```

One or more termination criteria may be used such as reaching a freezing temperature τ_f, exceeding a maximum number of unsuccessful function evaluations at a given temperature N_{max}, or attaining a minimum cost value.

III. Generating the fish model

This section describes a novel method for generating a fish-shape model to be used for model-based tracking. The shape model is based on the midline of the fish. There are several ways to generate the midline. In [14], the midline is found by projecting the top-view profile on a plane of orientation. In [12], [13], the midline is generated manually. The midline in our tracking system is generated automatically when the fish is in clear view of all cameras, i.e., when there are no occlusions and both head and tail are visible. The shape model is generated automatically from the midline using an iterated EKF. The relevant nomenclature is summarized in Table I.

TABLE I
Nomenclature

s	Midline coordinate, $s \in [0, 1]$
$\mathcal{E}(s)$	Elliptical cross section of fish body at s
$a(s)$	Semi-major axis of cross section at s
$b(s)$	Semi-minor axis of cross section at s
$d(s)$	Displacement of cross section along normal axis at s
c	Camera index, c=1,2,3
$\boldsymbol{f}(s)$	Midline at s in body-fixed reference frame
$\boldsymbol{S}$	The surface of a fish body
$\boldsymbol{h}$	Heading vector (orientation of head)
k	Time index, $k = 1, 2, \ldots$
L	A line in three dimensions
$\boldsymbol{m}(s)$	Midline in world reference frame at s
${}^cO$	Occluding contour in camera c
$\boldsymbol{p}$	Vector of polynomial coefficients
$\boldsymbol{g}$	Vertical axis in world frame
$\boldsymbol{t}(s)$	Tangent vector to the midline at s
$\boldsymbol{x}(s)$	Normal vector to the midline at s
$\boldsymbol{y}(s)$	Binormal vector to the midline at s
T	4×4 transformation matrix
${}^c\boldsymbol{u}$	Measurement in pixels in cth camera image
${}^c\hat{\boldsymbol{u}}$	Projected estimate in pixels in cth camera image
$\boldsymbol{X}_k$	State of a target at time k
$\boldsymbol{Z}_k$	Measurements at time k
$\mathcal{B}$	Body frame fixed to head
$\mathcal{C}$	Camera reference frame
$\mathcal{W}$	World reference frame

A. Shape representation using the midline

For the purpose of model generation and tracking we make the following assumptions about fish motion observed in our

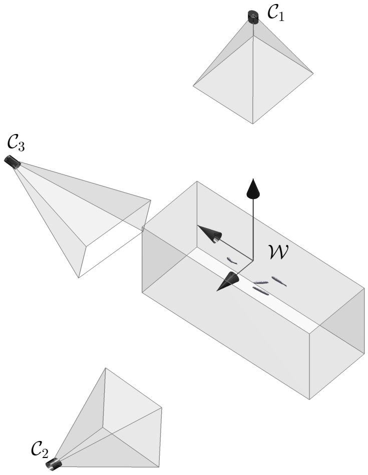


Fig. 2. Camera views $\mathcal{C}_1, \mathcal{C}_2$, and $\mathcal{C}_3$, and world frame $\mathcal{W}$. Cameras $\mathcal{C}_1$ and $\mathcal{C}_2$ are used for tracking; Camera $\mathcal{C}_3$ is used for validation purposes.

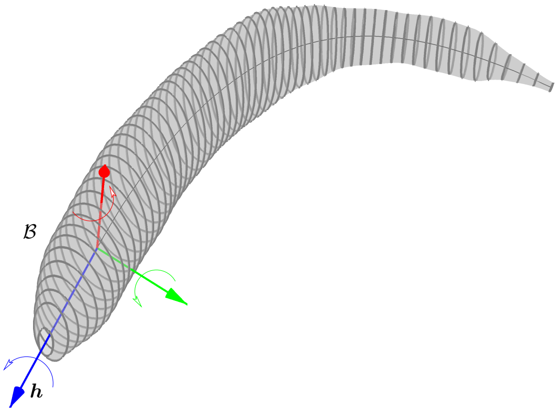


Fig. 3. The body frame $\mathcal{B}$ is fixed to the head with the heading vector $\boldsymbol{h}$ pointing towards the tip of the nose. The pitch (green), roll (blue), and yaw (red) axes complete the frame.

experiments:

A1) The fish in our tracking experiments bend laterally [14].

A2) The fish in our tracking experiments turn and pitch, but rarely roll.

A3) The portion of the body from the eyes to the nose (the head) does not bend.

A single fish is characterized by the position of the head, the orientation of the head (the heading vector), and the midline. The midline is a curve that runs from the head to the tail. A surface is generated around the midline to approximate the shape. We define the shape locally using a body-fixed reference frame $\mathcal{B}$. The origin of frame $\mathcal{B}$ is the center of the head with one axis in the direction of the nose. The heading $\boldsymbol{h} \in \mathbb{R}^3$ is a unit vector pointing from the center of the head to the tip of the nose (see Fig. 3). Based on assumption (A2), the body-frame axes are completed by performing the cross product of the vertical $\boldsymbol{g} \triangleq \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$ with the heading $\boldsymbol{h}$ to get the pitching axis, followed by the cross product between the heading and the pitching axis to get the yaw axis. Given the position of the head $\boldsymbol{r} \in \mathbb{R}^3$, the complete body frame in the world-frame coordinates can be represented by the transformation ${}^{\mathcal{W}}T_{\mathcal{B}} = \begin{bmatrix} \boldsymbol{h} & \boldsymbol{g} \times \boldsymbol{h} & \boldsymbol{h} \times (\boldsymbol{g} \times \boldsymbol{h}) & \boldsymbol{r} \\ 0 & 0 & 0 & 1 \end{bmatrix}$.

The midline is parameterized in the body frame by $\boldsymbol{f}(s) = \begin{bmatrix} f_1(s) & f_2(s) & f_3(s) \end{bmatrix}^T$, where $s \in [0,1]$. We assume the functions $f_i(s)$ are differentiable, which permits us to define an orthonormal frame at each point s on the midline. We use this frame to define the body cross section at s.

To allow for up to two inflection points and the possibility of a C-start or S-start, we model $f_1(s)$ and $f_2(s)$ as quadratic and quartic polynomials, respectively. We have

$$\begin{aligned} f_1(s) &= p_1 s + p_2 s^2 \\ f_2(s) &= p_3 s^2 + p_4 s^3 + p_5 s^4 \\ f_3(s) &= 0, \end{aligned} \tag{3}$$

where $\boldsymbol{p} = \begin{bmatrix} p_1, \ldots, p_5 \end{bmatrix}^T$ are the polynomial coefficients. The midline is represented in world-frame coordinates using transformation ${}^{\mathcal{W}}T_{\mathcal{B}}$, i.e,

$$\begin{bmatrix} \boldsymbol{m}(s) \\ 1 \end{bmatrix} = {}^{\mathcal{W}}T_{\mathcal{B}} \begin{bmatrix} \boldsymbol{f}(s) \\ 1 \end{bmatrix}. \tag{4}$$

The midline $\boldsymbol{m}(s)$ is projected onto the image by perspective projection, which uses the camera calibration parameters [35]. The projected midline ${}^c\hat{\boldsymbol{u}}(s)$ on camera c is [36]

$${}^c\hat{\boldsymbol{u}}(s) = \begin{bmatrix} \frac{{}^c w_1}{{}^c w_3} & \frac{{}^c w_2}{{}^c w_3} \end{bmatrix}^T,$$

where ${}^c\boldsymbol{w}(s) = {}^cP\boldsymbol{m}(s)$ and ${}^cP$ is the camera projection matrix [36].

To automatically generate the midline, we locate the head, nose, and tail of the fish from the top view (camera 1) based on the following observations: (a) the center of the head is the center of the largest circle that fits inside the silhouette; (b) the nose is the highest curvature point on the portion of the occluding contour near the head; and (c) the curvature of the occluding contour is highest at the tail. (Curvature, defined in Section V-C, represents the degree of bending.)

The location of the nose expressed in pixels in camera 1 is denoted by ${}^1\boldsymbol{u}_n$, the tail by ${}^1\boldsymbol{u}_t$, and the center of the head by ${}^1\boldsymbol{u}_h$. The distance of a point on the silhouette ${}^1\boldsymbol{u} \in \mathbb{R}^2$ from any point on the projected curve ${}^1\hat{\boldsymbol{u}}(s) \in \mathbb{R}^2$ is given by $\|{}^1\boldsymbol{u} - {}^1\hat{\boldsymbol{u}}(s)\|$. The side views (cameras 2 and 3) give orientation information as well as position information. Let ${}^cl \triangleq ({}^cl_m, {}^cl_r)$ be a line in camera c, where ${}^cl_m$ is the slope and ${}^cl_r$ is the intercept with the vertical axis of the image plane. A least-squares fit on the silhouette in camera c establishes a line from the head to the tail. The body frame is oriented so that the heading is aligned with this line in the side view and with the vector from the head to the nose in the top view. The head and nose are marked in the top view. We use the following additional constraints in the side view to build the body frame:

$$\begin{aligned} {}^c\hat{u}_{2,h} &= {}^cl_m {}^c\hat{u}_{1,h} + {}^cl_r \\ {}^c\hat{u}_{2,n} &= {}^cl_m {}^c\hat{u}_{1,n} + {}^cl_r, \end{aligned}$$

where $c \in \{2, 3\}$. We solve these constraint equations in either one of the side cameras for the position of the head $\boldsymbol{m}(0) \in \mathbb{R}^3$ and nose. We complete the body frame by applying the no-roll assumption (A2).

The estimated midline parameters $\hat{\boldsymbol{p}}$ are found using a nonlinear cost function that measures the distance of the silhouette to the midline. Let q_i^* be the distance of the point ${}^1\boldsymbol{u}_i$ in the top-view silhouette to the closest point on the projected midline ${}^1\hat{\boldsymbol{u}}(s)$. The midline parameters $\hat{\boldsymbol{p}}$ are estimated by solving

$$\begin{aligned} &\hat{\boldsymbol{p}} = \operatorname*{argmin}_{\boldsymbol{p}} \sum_i q_i^*, \text{ where} \\ &q_i^* = \min_s \|{}^1\boldsymbol{u}_i - {}^1\hat{\boldsymbol{u}}(s)\| \text{ subject to} \\ &{}^1\hat{\boldsymbol{u}}(1) = {}^1\boldsymbol{u}_t. \end{aligned} \tag{5}$$

We minimize (5) it by applying a two-stage optimization process consisting of simulated annealing followed by a quasi-Newton line search [37]. Once a midline is estimated, a surface is generated around it to create a shape model as described next.

B. Generating a shape model

We model the fish cross section at point s on the midline by an ellipse $\mathcal{E}(s)$ in the plane that is normal to the midline at s. We compute the ellipse planes at each point using curve framing [38]. The tangent $\boldsymbol{t}(s)$ to the midline at point s forms an axis of a local orthogonal frame $\begin{bmatrix}\boldsymbol{x}(s) & \boldsymbol{y}(s) & \boldsymbol{t}(s)\end{bmatrix}$. The local frame at each point on the midline is completed as follows: the normal axis $\boldsymbol{x}(s)$ is $\boldsymbol{x}(s) = \boldsymbol{g} \times \boldsymbol{t}(s)$ and the binormal is $\boldsymbol{y}(s) = \boldsymbol{t}(s) \times \boldsymbol{x}(s)$ (see Fig. 4). A point on the cross section $\mathcal{E}(s)$ can be represented in the world frame $\mathcal{W}$ using the transformation matrix

$${}^{\mathcal{W}}T_{\mathcal{E}} = \begin{bmatrix} \boldsymbol{x}(s) & \boldsymbol{y}(s) & \boldsymbol{t}(s) & \boldsymbol{m}(s) \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{6}$$

where

$$\boldsymbol{t}(s) = \begin{bmatrix} \frac{\partial m_1}{\partial s} & \frac{\partial m_2}{\partial s} & \frac{\partial m_3}{\partial s} \end{bmatrix}^T.$$

In order to generate the body surface, we estimate the major $a(s)$ and minor $b(s)$ axes of each elliptical cross section. We include a parameter, $d(s)$, that allows us to displace the ellipse along $\boldsymbol{y}(s)$. Using candidate values for $a(s), b(s)$, $d(s)$, and the transformation matrix above, we scale and transform the cross section $\gamma(u) = \begin{bmatrix}\cos(u) & \sin(u) & 0\end{bmatrix}^T$, where $u \in [0, 2\pi]$ [30], [39]. The transformation is defined as (see (2))

$$\delta(\gamma, s) = M(s)\gamma + \boldsymbol{T}(s), \tag{7}$$

where

$$\begin{aligned} M(s) &= \begin{bmatrix}\boldsymbol{x}(s)a(s) & \boldsymbol{y}(s)b(s) & \boldsymbol{0}_{3\times 1}\end{bmatrix} \\ \boldsymbol{T}(s) &= \boldsymbol{m}(s) + \boldsymbol{y}(s)d(s). \end{aligned} \tag{8}$$

The curve $\boldsymbol{m}(s)$ is formed using (4). Substituting (8) into (7), we obtain the surface

$$\begin{aligned} \boldsymbol{S}(s, u) \triangleq \delta(\gamma(u), s) = \boldsymbol{m}(s) + a(s)\cos(u)\boldsymbol{x}(s) + \\ (b(s)\sin(u) + d(s))\boldsymbol{y}(s), \end{aligned} \tag{9}$$

where $s \in [0, 1]$ and $u \in [0, 2\pi]$.

In order to generate an accurate surface model for each fish, we measure the values $a(s), b(s)$ and $d(s)$ using the top-view and side-view observations. These values are the state variables in the model-estimation process. Each measurement in this process is the length of the line segment contained in the occluding contour and normal to the midline (see Fig. 4).

We substitute $u = \{0, \pi\}$ in equation (9) to produce the end points of the major axis, $a(s)$; $u = \{\pi/2, 3\pi/2\}$ produces the values for the minor axis, $b(s)$, and $d(s)$. A perspective projection of a surface point $\boldsymbol{S}(s, u)$ on camera c is denoted by ${}^c\boldsymbol{S}(s, u)$. The measurement model is

$$\begin{aligned} p_a(s) &= \frac{\|{}^1\boldsymbol{S}(s, 0) - {}^1\boldsymbol{S}(s, \pi)\|}{2} + e_1 \\ p_b(s) &= \frac{\|{}^2\boldsymbol{S}(s, \pi/2) - {}^2\boldsymbol{S}(s, 3\pi/2)\|}{2} + e_2 \\ p_d(s) &= \left\| \frac{{}^2\boldsymbol{S}(s, \pi/2) + {}^2\boldsymbol{S}(s, 3\pi/2)}{2} - {}^2\hat{\boldsymbol{u}}(s) \right\| + e_2, \end{aligned} \tag{10}$$

where $p_a(s), p_b(s)$ and $p_d(s)$ are the measurements of $a(s), b(s), d(s)$ in the respective camera views, and e_1 and e_2 are the measurement noise in cameras 1 and 2, respectively. [1]

We use an iterated EKF to update our estimates. (An iterated EKF updates the estimate about the last computed value to minimize the measurement error.) A requirement for generating a reliable model is that we have a clear view of the fish including its nose and tail in all camera views at least once. The EKF is initialized by selecting all fish in each camera. Once the ellipse sizes are estimated, we can use them to generate a shape in combination with the state of the fish $\boldsymbol{X} = \begin{bmatrix}\boldsymbol{r}^T & \boldsymbol{h}^T & \boldsymbol{p}^T\end{bmatrix}^T \in \mathbb{R}^{11}$, where $\boldsymbol{r} = \boldsymbol{m}(0)$.

IV. Reconstructing fish motion

In this section we describe the steps for tracking individual fish after a model is generated. We first describe the metric used to associate target estimates to measurements, then present the objective function used to estimate the position, orientation, and shape trajectories.

The tracking algorithm associates the silhouette of a blob in a camera image to a target based on the proximity of the silhouette to the target's projected midline. Once a match is made, we use the estimated three-dimensional model to again project the boundary of the fish body (i.e., the occluding contour) onto a camera plane. This occluding contour is compared with the silhouette boundary in multiple views while varying the position, orientation, and shape until a best fit is obtained. We use a numerical optimization algorithm to find the best fit.

A. Finding measurement-target associations

In a multi-target tracking experiment, before we update a target estimate using a new measurement, we must first associate the measurement with a target. We use nearest-neighbor matching, which associates a measurement to a target based on

[1] Note that the above measurement model assumes that the occluding contour of a fish is a projection of the extreme ends of the elliptical cross sections. Since the camera distance (1 m) is much larger than the fish cross section (2.5 cm), this assumption introduces only sub-pixel measurement error.

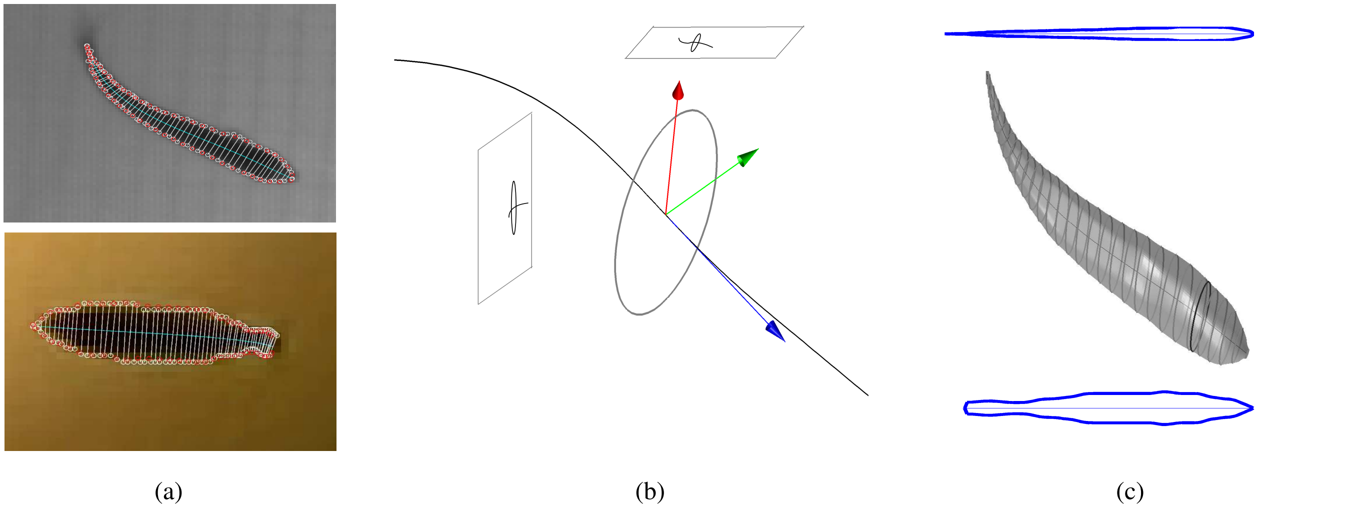


Fig. 4. Generating the fish shape model. (a) Midline fits and occluding contours in the top view and side view are used to generate a midline in three-dimensions. The white circles are the measurements and red circles are the projected estimates of the end points of the ellipse axes. (b) Cross-sectional ellipse normal to the midline. (c) The top profile and side profile are used to generate the final surface. (The black ellipse partitions the head and rest of the body.)

a generalized distance metric. The Euclidean distance between centroid positions may not provide an accurate association when the fish are close to one another, so we establish another metric described here.

The measurements in our case are silhouettes on a camera frame. Let the set of measurements on a camera frame be indexed by j. $\boldsymbol{Z}^j$ denotes a silhouette on the camera frame. The points in a silhouette are indexed by i, i.e, $\boldsymbol{u}_i^j \in \boldsymbol{Z}^j$. Note that $\boldsymbol{u}_i^j \in \mathbb{R}^2$ is measured in pixels. To match a silhouette with a target, we project the midline from each target onto the camera image plane. We then assign a silhouette to the target if it is the "closest" silhouette to the midline. The generalized distance metric computes the sum of the minimum distance of each point on the midline to a silhouette. Let ${}^c\hat{\boldsymbol{u}}_t(s)$ denote the projected midline of target t. The measurement j_t associated to target t in frame c is computed by solving

$$\begin{aligned} j_t &= \operatorname*{argmin}_j \sum_s q_s^* \text{ where} \\ q_s^* &= \min_i \|{}^c\boldsymbol{u}_i - {}^c\hat{\boldsymbol{u}}_t(s)\|. \end{aligned} \tag{11}$$

Note that in (11) the minimum distance from the midline is computed. This is because we are not attempting to fit the midline to a silhouette, but rather to find how far it is from a given silhouette. In the case of an occlusion, two targets are assigned the same silhouette. The search space is automatically increased so that we now fit multiple shape projections to the same silhouette.

B. Shape-matching cost function

Once a model is generated we produce a three-dimensional line from each point on the occluding contour O. The distance of each line to the model surface $\boldsymbol{S}$ is used to optimize the state estimate [40]. We represent a line L in three dimensions by Plücker coordinates [40]. The advantage of this representation is that it defines a line uniquely and its distance to a point is a straightforward operation. Let $L \triangleq \begin{bmatrix}\boldsymbol{l}_v^T & \boldsymbol{l}_m^T\end{bmatrix}^T$, where $\boldsymbol{l}_v \in \mathbb{R}^3$ is the unit vector representing the direction of the line and $\boldsymbol{l}_m = \boldsymbol{l}_r \times \boldsymbol{l}_v$ is the moment of any point $\boldsymbol{l}_r \in \mathbb{R}^3$ on the line. The distance of point $\boldsymbol{r}$ from the line L is given by $\|\boldsymbol{r} \times \boldsymbol{l}_v - \boldsymbol{l}_m\|$. The cost function is a measure of the total distance of a surface from an occluding contour. We denote a point on the surface $\boldsymbol{S}$ by $\boldsymbol{S}_i$. The state estimate $\hat{\boldsymbol{X}}$ is obtained by solving

$$\begin{aligned} \hat{\boldsymbol{X}} &= \operatorname*{argmin}_{\boldsymbol{X}} \sum_{c\in(1,2)} \sum_{o\in{}^cO} {}^cD_o^* + g(\Delta l) \text{ where} \\ {}^cD_o^* &= \min_{i\in\boldsymbol{S}} \|\boldsymbol{S}_i \times {}^c\boldsymbol{l}_{v,o} - {}^c\boldsymbol{l}_{m,o}\|. \end{aligned} \tag{12}$$

$g(\Delta l)$ is a non-decreasing function of Δl, the difference in length between the midline as computed from shape-estimation and from the candidate state $\boldsymbol{X}$. In Section V we choose $g(\Delta l) = K_g\|\Delta l\|^2$, where $K_g > 0$.

We use simulated annealing to search the state space. New points are generated at each step of the optimization algorithm using Gaussian disturbance $\boldsymbol{w}$ with known covariance. Unlike an iterative closest-point algorithm [41], we do not perform a prior association between the measurements and the surface. This permits larger variation in pose and shape, which is common during fast starts.

C. Filtering the state estimates

The optimization output is rarely smooth because errors in the measurements are absorbed into the estimates. We smooth the estimates by passing the output state through a Kalman filter. Fish movement comprises change in position, orientation, and shape. We model velocity and heading vector as being subject to Gaussian disturbance

$$d\dot{\boldsymbol{r}} = d\boldsymbol{w}_r,\ d\boldsymbol{h} = d\boldsymbol{w}_h, \tag{13}$$

where $\boldsymbol{w}_r, \boldsymbol{w}_h \in \mathbb{R}^3$ indicate white noise processes.

The shape consists of the curve parameters $\boldsymbol{p} = \begin{bmatrix}p_1, \ldots, p_5\end{bmatrix}^T$. In a straight midline, p_1 represents the length of

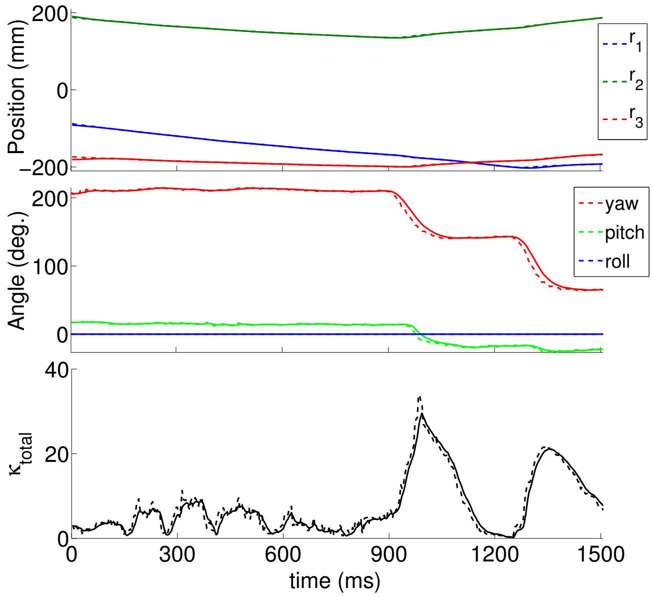


Fig. 5. Time series plots of position, orientation, and total curvature for a single fish. The plots are shown before filtering (dashed lines) and after filtering (solid lines). The two peaks in the total curvature correspond to turns. κ_{total} is defined in Section V-C.

TABLE II
PARAMETER VALUES USED FOR TRACKING

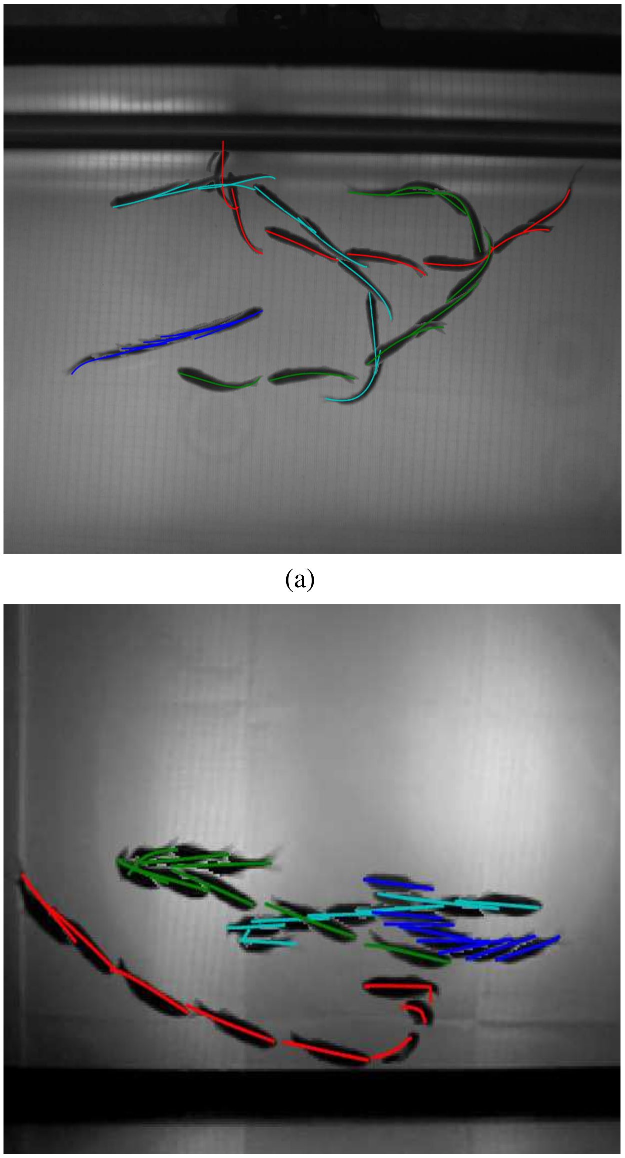
Parameter	Value	Description
α	0.05	Background update coefficient (initial)
α	0.0001	Background update coefficient (final)
λ	5	Coefficient of decay for midline parameters $\boldsymbol{p}$
τ	1	Starting temperature for simulated annealing
e_1	1	Noise variance in top view (pixels)
e_2	2	Noise variance in side view (pixels)
$\Sigma_{\boldsymbol{w}}$	1	Noise variance for generating new points in SA
K_g	10	Weighting factor in shape-matching cost function
τ_f	10^{-6}	Freezing temperature for simulated annealing
K_c	0.9	Cooling coefficient for cooling schedule
N_{max}	400	Max. unsuccessful evaluations at a temperature

the fish and $p_2, \ldots, p_5$ are all zero. A bent midline corresponds to non-zero values in $p_2, \ldots, p_5$. We model the fish as having constant length; the midline tends to straighten out after being bent. We therefore model changes in p_1 using Gaussian noise $w_{p,i}$, and model $p_2, \ldots, p_5$ as exponentially decaying variables with rate $\lambda > 0$, i.e,

$$\begin{aligned} dp_1 &= dw_{p,1} \\ dp_i &= -\lambda p_i dt + dw_{p,i}, \;\; i = 2, \ldots, 5. \end{aligned} \tag{14}$$

V. Experimental validation

A. Materials and setup


(a)

(b)

Fig. 6. A multi-exposure image with estimated midlines projected on the image plane. (a) Top and (b) side views of four fish.

In order to test our tracking framework, we filmed trials of one, two, four, and eight giant danio (*Danio aequipinnatus*) in a 0.61 m × 0.30 m × 0.40 m (24"×12"×16"), 20 gallon tank. Each trial lasted for 1–3 seconds. Three cameras were used to film the fish (see Fig. 2). Two cameras were used for tracking and the third camera was used for validation. A DRS Lightning RDT high-resolution camera was placed above the tank to capture the top view at 250 frames per second (fps) and 1280 × 1024 pixel resolution. Two Casio EX-F1 Pro cameras were placed orthogonal to each other facing the tank sides. These cameras captured images at 300 fps and 512 × 384 pixel resolution. To ensure an adequately lit background, the remaining three sides of the tank were back-lit by a 150W fluorescent light source diffused by 1/4 stop with a diffuser fabric. Videos from the three cameras were synced by marking a frame in each video with a distinct common event. Simultaneous events during a trial were generated in the field of view of all three cameras by a string of flashing LEDs. The full videos were then synced and verified using a custom Linux shell script. (Every fifth frame in the 250 fps video was repeated.)

At the beginning of each experiment, a short video of the tank was recorded without any fish, so that we could model

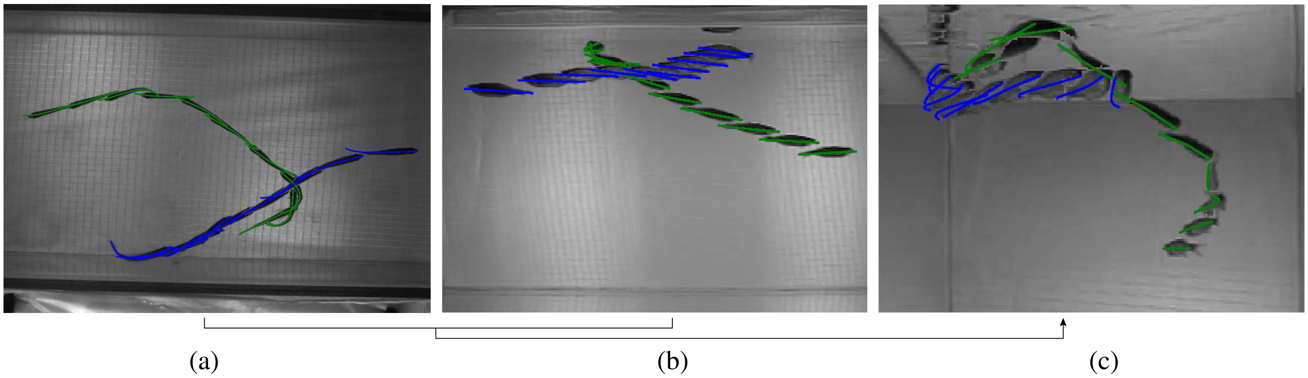


Fig. 7. Tracking validation using an independent camera. The shape estimated from the (a) top and (b) side camera is projected onto a (c) multi-exposure image from the independent camera.

the background for background subtraction. Each tracking sequence starts with a set of background images, wherein the background is modeled as a running average with a tuning parameter ${}^{c}\alpha$ [43]:

$$ {}^{c}B_{k+1} = {}^{c}B_k(1 - {}^{c}\alpha) + {}^{c}\alpha {}^{c}I_{k+1}, \tag{15} $$

where ${}^{c}B_0$ is the first background image and ${}^{c}I_k$ is the current image of camera c. The value of ${}^{c}\alpha$ was kept high initially to model lighting fluctuations and was lowered when there were fish present. (See Table II for parameter values used for tracking.)

Camera calibration was performed using the MATLAB calibration toolbox [44]. A planar checkerboard was filmed underwater at different orientations inside the tank. Extrinsic calibration was performed by moving the checkerboard between the cameras and propagating the extrinsic parameters between overlapping camera views until all camera positions and orientations were known with respect to the world frame. The reprojection error during calibration for each camera was in subpixels. In three dimensions, the error was computed by comparing the known distance between checkerboard points (ranging between 30 mm and 210 mm apart) with the distance between estimated position. The average error over 50 such observations was 0.7 ± 0.37 mm. The world frame was chosen to be directly below the top camera such that the vertical axis pointed up (see Fig. 2). The top-view camera and the tank were aligned using a bubble level.

Once the calibration was performed, fish were introduced into the tank from a separate tank in sets of 1, 2, 4 and 8. Three trials were conducted for each set. Filming was started approximately ten minutes after the fish were introduced. The input to the tracking system was a set of synced frames from each camera (top and side) and the calibration parameters for each camera. The output is a time series of the state vector $\boldsymbol{X}$ for each fish. The number of fish was constant during each trial.

B. Validation of tracking accuracy

Results for the tracking system are reported here for five out of the twelve trials. In every trial, we were able to track multiple fish shapes even during occlusions. The maximum density of the fish schools that we tracked was one fish per 2.5 gallons. (The actual density was higher because the fish schooled in only a fraction of the tank volume.) We used two methods to determine the accuracy of our tracking algorithm. First, the estimated shape and track reconstruction were verified using an independent camera. Fig. 7 illustrates the accuracy of the tracker using the projected estimate on the third camera. Second, we randomly selected a set of frames across multiple videos and manually marked ten control points along the midline in the top view. The midline was then manually generated by interpolating a curve between the ten marked points. The orthogonal distance between each point on the estimated midline and the manually generated midline was computed at each point. Fig. 8 depicts the average, maximum, and minimum error on the midline. Comparing the manually generated midline and tracked midline in the top view for a single fish shows a maximum average error of five pixels at the tip of the tail. The tail error is primarily due to the inconsistent appearance of the semi-transparent caudal fin in the silhouette measurements.

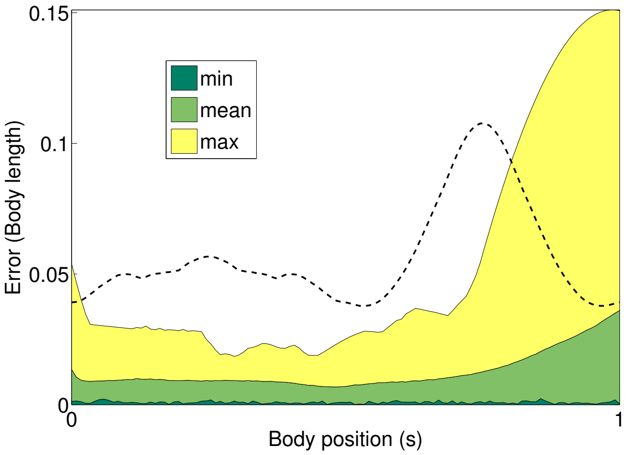


Fig. 8. Error for midline fit. The midline was manually selected on a random set of 100 top-view frames. The distance between the projected estimate and the manually generated midline as measured in the top camera frame. For comparison with previous work, we also computed the mean error (dashed line) for a fish shape modeled as a bent ellipsoid [21].

Occlusions of two and three fish were tracked reliably as evidenced by Fig. 10. Since the tracking process depends on the silhouettes in each camera frame to estimate the fish position, orientation, and shape, the tracking accuracy is affected by the number of fish in an occlusion. In our setup, with the low-resolution side cameras, we found loss of accuracy in occlusions with four or more fish (See Fig. 11). There were no data association errors, although these are expected for dense occlusions. We intend to address this problem by increasing the camera resolution and number so that the views with the fewest occlusions can be used to estimate shape.

C. Preliminary analysis of fast-start behavior

The shape-tracking system described in this paper yields a new opportunity to study fish behavior. The full-body reconstruction at every step allows one to automatically detect and quantify fast-start behavior, which we are doing in ongoing work outside the scope of this paper. Fig. 9 compares the curvature profile for a coasting motion with the profile for a fright response. We compute curvature and total curvature from the midline $\boldsymbol{f}(s)$ as [45]

$$\kappa = \frac{|f_1' f_2'' - f_2' f_1''|}{(f_1'^2 + f_2'^2)^{3/2}} \text{ and } \kappa_{\text{total}} = \int_0^1 \kappa(s) ds. \qquad (16)$$

In the first case, the fish was filmed without any disturbance. The second case is a midline reconstruction of a single fish from a multi-fish trial during which the fish was startled by a visual stimulus.

When no fright stimulus was presented, the curvature is high towards the tail. A coasting turn takes more than one hundred milliseconds and the curvature profile is flat. In the case of a fright response (an S-start), high curvature appears along the midline. The turn occurs in approximately 40 ms and appears as a dark band at 450 ms. A thin dark region near body length 0.9 appears in the curvature plots due to the combined effect of tail beat movement and inaccuracy in the tail reconstruction due to inconsistent appearance of the caudal fin. The three-dimensional reconstruction of each of these turns shows the distance travelled by each fish during the turn. The coasting fish travels 54 mm in 500 ms where as the startled fish travelled 160 mm in the same time.

VI. Conclusions and ongoing work

In this paper we describe a three-dimensional tracking framework for reconstructing the swimming kinematics of individual fish in a school and present results for schools with densities greater than one fish per 2.5 gallons. We used model-based tracking to estimate fish shape with multiple camera views. Using elliptical cross sections on the midline we automatically generate a shape model that is used to track the fish in three dimensions. A cost function that measures the distance between a three-dimensional surface and occluding contours on multiple camera planes is used in a simulated annealing algorithm to estimate shape at each time step. The output of the simulated annealing algorithm is passed through a Kalman filter to further smooth the estimates.

Tracking results with up to eight fish are shown and validated. The validation is performed using an independent camera. We are currently using this system to study fish schooling behavior by investigating fast-start time signatures in the curvature profiles.

The inaccuracies in the tracker result primarily from (a) the modeling assumption that the fish midline lies on an inclined plane; (b) dense occlusions, during which the limited resolution of the cameras make it difficult to resolve the silhouettes into individual shapes; and (c) the curve parameterization which may be insufficient to represent complex curves. The accuracy of the tracker can be further improved by segregating the head and orientation tracking from shape tracking when there are no occlusions. A particle filter may be run to track the head and orientation while simulated annealing can be used to estimate shape.

Inaccuracies may also result due to refraction between air and water. In the case of our setup where the camera image plane was parallel to the water surface and centered with respect to the face of the tank, errors due to refraction were low (Section V-A), however, mounting the cameras at an angle to the water surface would require compensation for refraction effects.

As part of ongoing work we are improving the tracking speed. The tracking software has been developed in MATLAB where it takes an average of four seconds per fish per frame on a 2 GHz CPU with 4 GB of memory (The tracker is used as a post-trial analysis tool.) The majority of the computation time is spent in the optimization step to find the shape fit. During occlusions, the search space increases n-fold, where n is the number of fish involved in an occlusion. The average time to resolve a two-fish occlusion was twelve seconds per fish. A realistic goal is to be able to track a single fish in 300 frames within 60 seconds. That would allow a user to study the results and make any changes for the next trial within several minutes. A first step in this direction would be to parallelize the optimization algorithm. For example, the annealing particle filter [46] runs a simulated annealing algorithm at multiple points in the state space to eventually reach the global minimum. Other variants of simulated annealing algorithm include modifying the sampling distribution and cooling schedule [47].

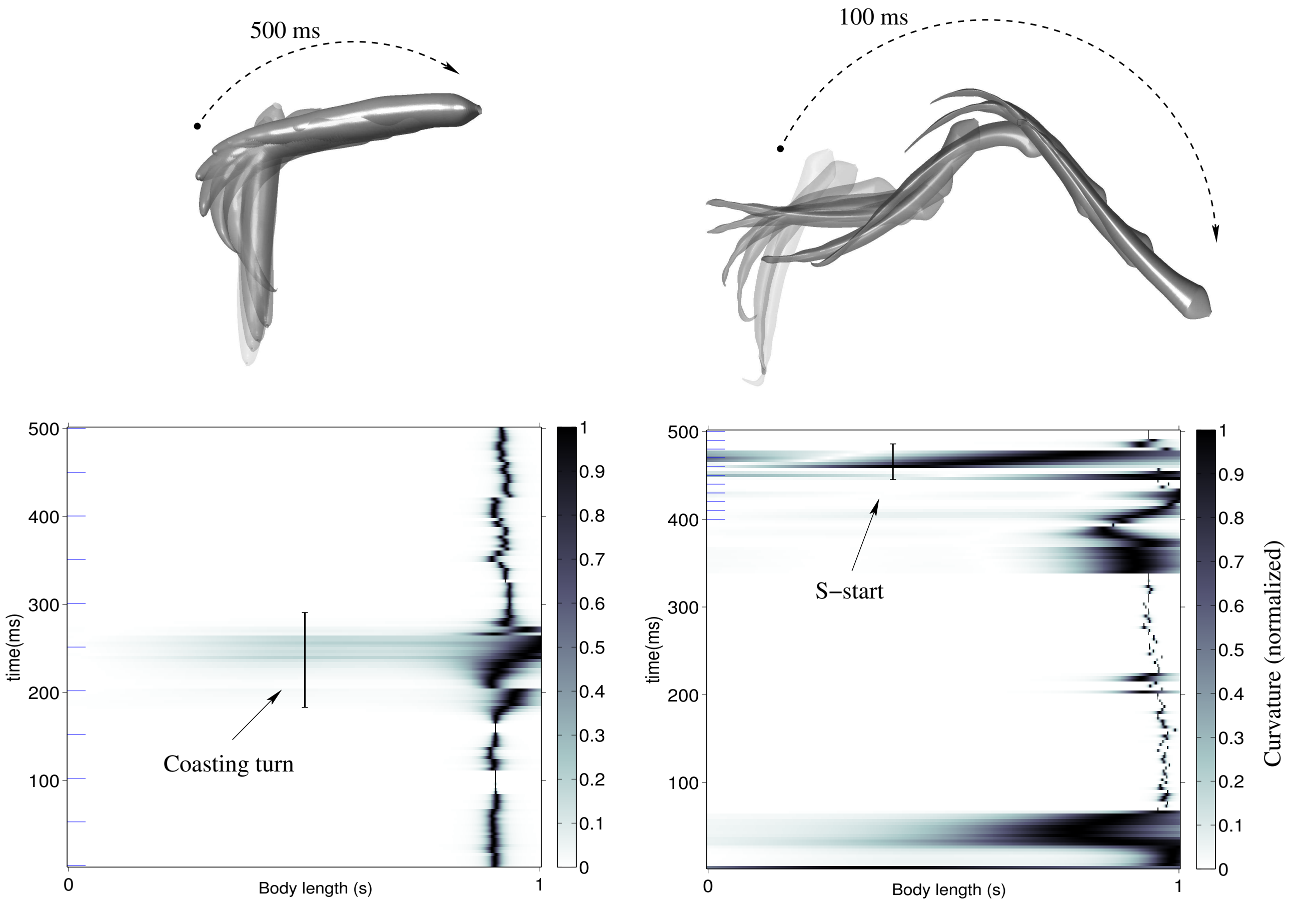
Acknowledgment

We are grateful to Arthur Popper for providing us with the lab space to perform these experiments and to members of the Popper Lab for taking care of the fish. We appreciate the help from Jennifer Lun and Amanda Chicoli during our experiments. We would also like to thank Eric Tytell and Avis Cohen for loaning us the high-speed camera, and Sheryl Coombs for her collaboration on the experiment design. This material is based upon work supported by the National Science Foundation under Grant No. CMMI0954361.

References

[1] J. K. Parrish and W. M. Hammer, *Animal groups in three dimensions*. Cambridge University Press, 1997.

[2] D. Sumpter, J. Buhl, D. Biro, and I. Couzin, “Information transfer in moving animal groups,” *Theory in Biosciences*, vol. 127, no. 2, pp. 177–186, 2008.



(a)

(b)

Fig. 9. Curvature profile comparison of a coasting turn and a fast-start using the three-dimensional reconstruction at fixed intervals (blue ticks). Curvature profile of fish midline during (a) a coasting turn with no fright stimulus and (b) a fast-start turn. The curvature is normalized through the body length at each time step. The dotted arc marks the beginning and end of the turn. The time in ms denotes the duration of the turn.

[3] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel Type of Phase Transition in a System of Self-Driven Particles," *Physical Review Letters*, vol. 75, pp. 1226–1229, Aug. 1995.

[4] A. Jadbabaie, J. Lin, and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," in *IEEE Conf. on Decision and Control*, Dec. 2002, pp. 988–1001.

[5] I. Couzin, J. Krause, N. Franks, and S. Levin, "Effective leadership and decision-making in animal groups on the move," *Nature*, vol. 433, no. 7025, pp. 513–516, 2005.

[6] J. Videler, *Fish swimming*. Springer, 1993.

[7] P. Domenici and R. W. Blake, "The kinematics and performance of fish fast-start swimming," *J.exp. Biol.*, vol. 200, no. 8, pp. 1165–1178, 1997.

[8] D. V. Radakov, *Schooling in the ecology of fish*. J. Wiley, 1973.

[9] N. Handegard, R. Patel, and V. Hjellvik, "Tracking individual fish from a moving platform using a split-beam transducer," *The Journal of the Acoustical Society of America*, vol. 118, pp. 2210–2223, 2005.

[10] S. V. Viscido, J. K. Parrish, and D. Grünbaum, "Individual behavior and emergent properties of fish schools: a comparison of observation and theory," *Marine Ecology Progress Series*, vol. 10.3354/meps273239, pp. 239–249, 2004.

[11] C. Schell, S. P. Linder, and J. R. Zeidler, "Tracking highly maneuverable targets with unknown behavior," *Proc. of the IEEE 92(3)*, pp. 558–574, 2004.

[12] E. Tytell and G. Lauder, "The C-start escape response of Polypterus senegalus: bilateral muscle activity and variation during stage 1 and 2," *Journal of Experimental Biology*, vol. 205, no. 17, pp. 2591–2603, 2002.

[13] E. Fontaine, D. Lentink, S. Kranenbarg, U. Muller, J. Van Leeuwen, A. Barr, and J. Burdick, "Automated visual tracking for studying the ontogeny of zebrafish swimming," *Journal of Experimental Biology*, vol. 211, no. 8, pp. 1305–1316, 2008.

[14] N. Hughes and L. Kelly, "New techniques for 3-D video tracking of fish swimming movements in still or flowing water," *Canadian Journal of Fisheries and Aquatic Sciences*, vol. 53, no. 11, pp. 2473–2483, NOV 1996.

[15] S. Dambreville, Y. Rathi, and A. Tannenbaum, "Tracking deformable objects with unscented Kalman filtering and geometric active contours," in *American Control Conference*, pp. 2856–2861, 2006.

[16] T. Chan and L. Vese, "Active contours without edges," *IEEE Transactions on image processing*, vol. 10, no. 2, pp. 266–277, 2001.

[17] T. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis," *Computer vision and image understanding*, vol. 104, no. 2-3, pp. 90–126, 2006.

[18] B. Stenger, P. Mendonca, and R. Cipolla, "Model-based 3D tracking of an articulated hand," *In Proc. IEEE Computer Vision and Pattern Recognition*, vol. 2, pp. 310–315, 2001.

[19] H. Sidenbladh, M. Black, and D. Fleet, "Stochastic tracking of 3D human figures using 2D image motion," *European Conference on Computer Vision*, pp. 702–718, 2000.

[20] D. Terzopoulos and D. Metaxas, "Dynamic 3D models with local and global deformations: deformable superquadrics," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13, pp. 703–714, 1991.

[21] S. Butail and D. A. Paley, "3D reconstruction of fish schooling kinematics from underwater video," in *IEEE Conference on Robotics and Automation*, 2010, pp. 2438–2443.

[22] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Computing Surveys (CSUR)*, vol. 38, no. 4, 2006.

[23] D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. on Automatic Control*, pp. 843–854, 1979.

[24] Y. Bar-Shalom, *Tracking and data association*. San Diego, CA, USA: Academic Press Professional, Inc., 1987.

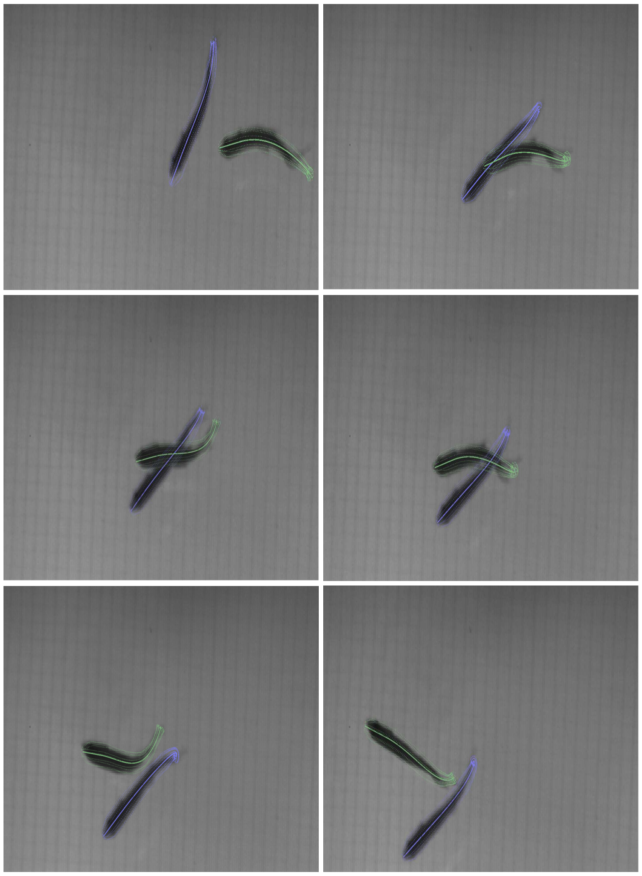


Fig. 10. Sequence of frames showing shape tracking during an occlusion. See video in supplementary material (also on http://youtu.be/SgDLNjA1MbU).

[25] I. J. Cox, "A review of statistical data association for motion correspondence," *Int. J. Comput. Vision*, vol. 10, no. 1, pp. 53–66, 1993.

[26] K. Branson, A. A. Robie, J. Bender, P. Perona, and M. H. Dickinson, "High-throughput ethomics in large groups of drosophila," *Nature Methods*, vol. 6, pp. 451–457, 2009.

[27] D. Grover, J. Tower, and S. Tavaré, "O fly, where art thou?" *Journal of the Royal Society Interface*, vol. 5, no. 27, p. 1181, 2008.

[28] A. Straw, K. Branson, T. Neumann, and M. Dickinson, "Multi-camera real-time three-dimensional tracking of multiple flying animals," *Journal of The Royal Society Interface*, 2010.

[29] Z. Khan, T. Balch, and F. Dellaert, "An MCMC-based particle filter for tracking multiple interacting targets," *Proc. European Conf. on Computer Vision (ECCV)*, pp. 279–290, 2004.

[30] J. Snyder and J. Kajiya, "Generative modeling: A symbolic system for geometric modeling," in *Proceedings of the 19th Annual Conference on Computer graphics and Interactive Techniques*, 1992.

[31] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb 2002.

[32] B. Bell and F. Cathey, "The iterated Kalman filter update as a Gauss-Newton method," *IEEE Trans. on Automatic Control*, vol. 38, no. 2, pp. 294–297, 1993.

[33] S. Kirkpatrick, "Optimization by simulated annealing: Quantitative studies," *Journal of Statistical Physics*, vol. 34, no. 5, pp. 975–986, 1984.

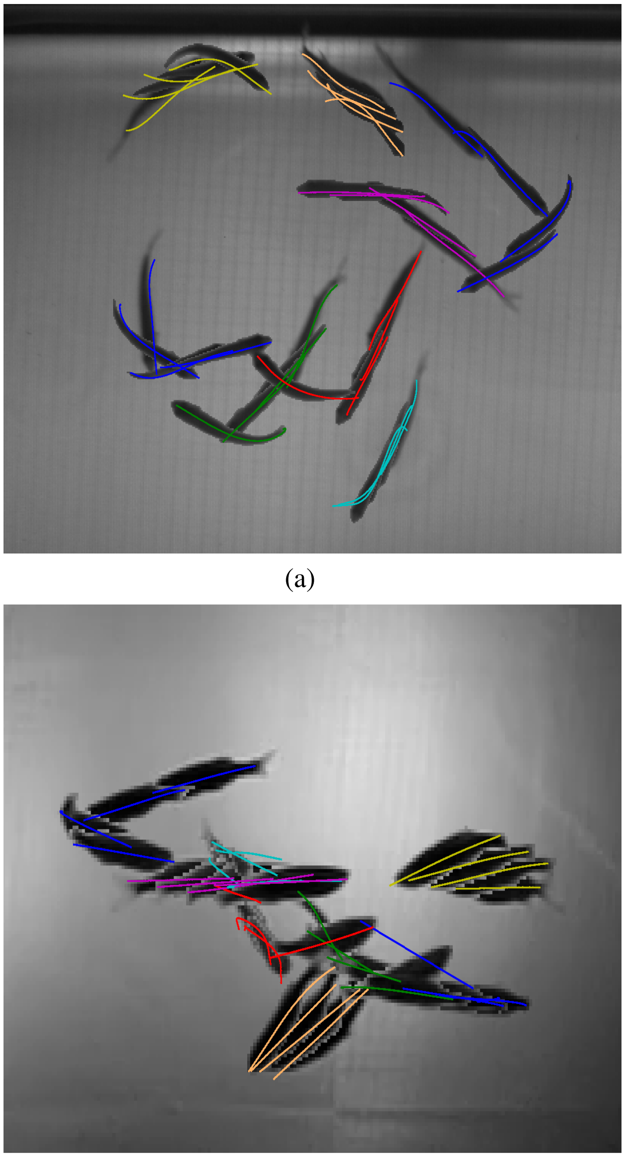
[34] J. Vandekerckhove (2010). General simulated annealing algorithm. MATLAB Central File Exchange. Retrieved Oct 14, 2010.

[35] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *International Conference on Computer Vision*, vol. 1, 1999, pp. 666–673.

[36] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, March 2004.

[37] J. Nocedal and S. Wright, *Numerical optimization*. Springer Verlag, 1999.

[38] A. Hanson, "Quaternion Frenet frames: Making optimal tubes and ribbons from curves," *Technical Report*, no. 1.111, 1993.



(a)

(b)

Fig. 11. Top and side views of eight fish tracked through 250 frames. Due to low resolution, the tracking accuracy is reduced (blue midline) in the side-view camera during dense occlusions.

[39] E. Fontaine, F. Zabala, M. Dickinson, and J. Burdick, "Wing and body motion during flight initiation in Drosophila revealed by automated visual tracking," *Journal of Experimental Biology*, vol. 212, no. 9, p. 1307, 2009.

[40] B. Rosenhahn, T. Brox, and J. Weickert, "Three-dimensional shape knowledge for joint image segmentation and pose tracking," *International Journal of Computer Vision*, vol. 73, no. 3, pp. 243–262, 2007.

[41] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. Third Int. 3-D Digital Imaging and Modeling Conf.*, 2001, pp. 145–152.

[42] X. Rong Li and V. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," *IEEE Trans. Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, Oct. 2003.

[43] M. Piccardi, "Background subtraction techniques: a review," *Int. Conf. on Systems, Man and Cybernetics*, vol. 4, pp. 3099–3104, Oct 2004.

[44] J.-Y. Bouguet. Camera Calibration Toolbox for Matlab. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/index.html

[45] F. Mokhtarian and A. K. Mackworth, "A theory of multiscale, curvature-based shape representation for planar curves," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 8, pp. 789–805, 1992.

[46] J. Deutscher and I. Reid, “Articulated body motion capture by stochastic search,” *International Journal of Computer Vision*, vol. 61, no. 2, pp. 185–205, 2005.

[47] Ingber, L., “Adaptive simulated annealing (ASA): Lessons learned,” *Arxiv preprint cs/0001018*, 2000.