3D shape reconstruction of semi-transparent worms

Thomas P. Ilett* Omer Yuval* Thomas Ranner* Netta Cohen*† David C. Hogg*†
University of Leeds, Leeds, United Kingdom

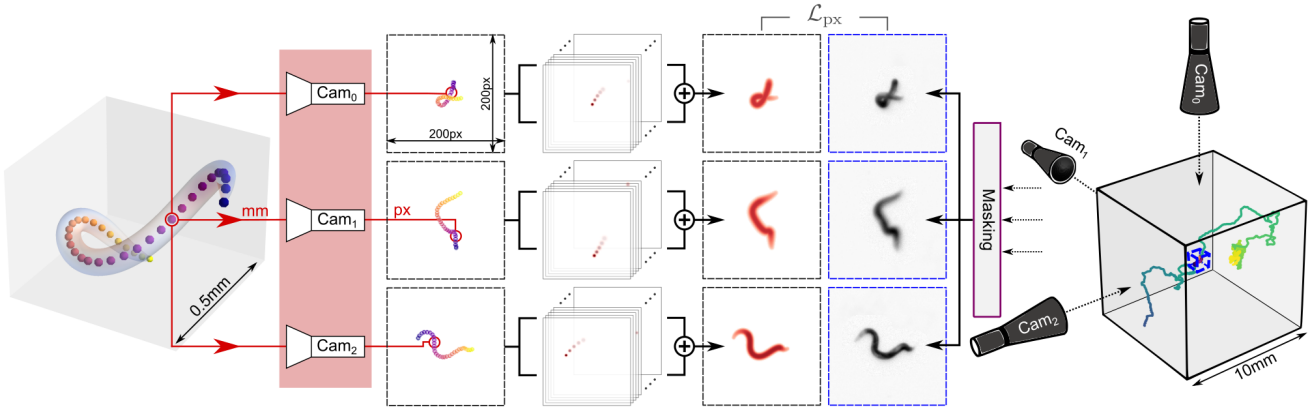

Figure 1. Posture reconstruction pipeline and imaging setup.

Abstract

3D shape reconstruction typically requires identifying object features or textures in multiple images of a subject. This approach is not viable when the subject is semi-transparent and moving in and out of focus. Here we overcome these challenges by rendering a candidate shape with adaptive blurring and transparency for comparison with the images. We use the microscopic nematode Caenorhabditis elegans *as a case study as it freely explores a 3D complex fluid with constantly changing optical properties. We model the slender worm as a 3D curve using an intrinsic parametrisation that naturally admits biologically-informed constraints and regularisation. To account for the changing optics we develop a novel differentiable renderer to construct images from 2D projections and compare against raw images to generate a pixel-wise error to jointly update the curve, camera and renderer parameters using gradient descent. The method is robust to interference such as bubbles and dirt trapped in the fluid, stays consistent through complex sequences of postures, recovers reliable estimates from blurry images and provides a significant improvement on previous attempts to track* C. elegans *in 3D. Our results demonstrate the potential of direct approaches to shape estimation in complex physical environments in the absence of ground-truth data.*

1. Introduction

Many creatures such as fish, birds and insects move in all directions to search and navigate volumetric environments. Acquiring 3D data of their motion has informed models of locomotion, behaviour and neural and mechanical control [3,22]. While technological advances have made the collection of large quantities of multi-viewpoint visual data more attainable, methods for extracting and modelling 3D information remain largely domain-dependant as few species share common geometric models or exist within the same spatial and temporal scales [4, 11, 14, 26, 37, 41, 50, 54, 65]. Furthermore, while humans and some domesticated animals [30,60] may act naturally while wearing special markers, marker-less observations of many species makes fea-

*{T.Ilett, O.Yuval, T.Ranner, N.Cohen, D.C.Hogg}@leeds.ac.uk
Funding This work was supported by University of Leeds and EPSRC.
Author contributions Conceptualisation, Methodology, Formal analysis, Investigation, Software, Visualisation: TPI. Data curation, Validation: TPI, OY. Writing: TPI (original), all (review and editing). Funding acquisition, Supervision: NC, DCH, TR. † Equal contribution.
Acknowledgements Additional thanks to Matan Braunstein (for help with Fig. 1), Robert I. Holbrook (data), Felix Salfelder (discussions and data), Lukas Deutz (discussions) and Jen Kruger (proof reading).
Data availability Supplementary movies are available here: https://doi.org/10.6084/m9.figshare.22310650.

ture extraction more challenging and means pose estimation generally lacks ground-truth data [48].

As a case study in marker-less 3D shape reconstruction, we consider *C. elegans*, a hair-thick, $\sim$1 mm long animal with a simple tapered cylinder shape, which can be constructed from a midline "skeleton". In the wild, *C. elegans* can be found in a wide range of complex 3D environments, *e.g.* decomposing organic matter, with continually changing physical properties [15, 17, 46]. However, to date, experiments have focused nearly exclusively on locomotion on a plane, limiting insight to the constrained, planar behaviours.

We obtained a large dataset (4 hours 53 minutes $\simeq$ 440,000 frames at 25Hz) of experimental recordings of individual worms moving freely inside a glass cube filled with a gelatin solution. The cube is positioned between three nearly-orthogonal static cameras fitted with telecentric lenses. Initial pinhole camera model parameter estimates are provided [45] but are imprecise and require continuous adjustment across the course of a recording to account for small vibrations and optical changes to the gel. We aim to simultaneously reconstruct a 3D shape and find corrected camera parameters to match these recordings in a process akin to bundle adjustment [56].

3D reconstruction typically involves the identification and triangulation of common features from multiple viewpoints or the synthesis of full images including texture and shading information to match given scenes [16, 21, 47, 66]. Imaging animals with length $\sim$1 mm requires sufficient magnification, but simultaneously capturing long-term trajectories up to 25 minutes requires a large volume of view (10-20 worm lengths per axis). As the worm explores the cube it frequently appears out of focus in one or more of the cameras. Air bubbles and dirt trapped in the gel along with old tracks are difficult to differentiate from the transparent worm, particularly at the tapered ends. Self occlusion invariably appears in a least one view, where hidden parts darken the foreground while the ordering of fore/back-parts is not discernible. As the semi-transparent and self-occluding subject moves in the volume, photometric information in one view bears little relevance to the appearance in the others making feature identification and photometric matching particularly challenging. We found that standard approaches may suffice for limited sub-clips, but lose parts of the object or fail catastrophically for much of the data and the solution requires a degree of adaptation.

We present an integrated "project-render-score" algorithm to obtain a midline curve for each image-triplet (Fig. 1). Discrete curve vertices are *projected* through a triplet of pinhole camera models, *rendered* to produce an image-triplet for direct comparison against the recorded images and *scored* according to their intersection with worm-like pixels in all three views. The differentiable renderer stacks 2D super-Gaussian blobs at the projected locations of each vertex to approximate the transparency along the worm, accounting for the variable focus and providing soft edges that direct the geometric model towards the midline. The scoring allows the detection of incongruities and keeps the curve aligned to the worm in all views. Regularisation terms ensure smoothness along the body and in time. Curve, camera and rendering parameters are jointly optimised using gradient descent to convergence. Once the worm shape has been resolved, it is generally only lost during image degradation or significant self-occlusions that make the posture unresolvable by eye.

In summary, our main contributions are:

- A robust pipeline for 3D posture reconstruction of a freely deforming semi-transparent object from noisy images.
- A novel viewpoint renderer to capture optical distortions and transparency.
- A feature-free bundle adjustment algorithm using direct image comparison and gradient descent.

2. Related work

Bundle adjustment (BA) is a procedure to jointly optimise 3D geometry and camera parameters [21, 56]. BA typically identifies common features of an object from multiple viewpoints in order to minimise a prediction error between projections of the corresponding 3D points and their 2D observations. BA is frequently used in conjunction with other methods to find camera parameters using multiple images of a 3D calibration object with known control points or for fine-tuning results [13, 23, 36, 40, 57, 59].

Feature detection converts photometric information into image coordinates. In BA, coordinates of common features are used to solve a geometric optimisation problem. Photometric bundle adjustment methods additionally require objects to have the same appearance in all views [12, 18]. Our method is entirely photometric, as such differing from BA. As our objects appear differently across views, all pixel information is used and the geometry is solved intrinsically.

Pose estimation Deep network approaches have proved well-suited to 2D human-pose estimation as they are potent feature extractors and large annotated training sets are available [1, 51, 55]. For 3D postures, ground truth multi-view datasets are less common. Recent progress [35] relies on end-to-end architectures [19, 27, 29, 32, 42, 61] or splitting the problem into 2D pose estimation and then constructing the 3D pose [10, 38]. Despite similar approaches used for non-human pose estimation, the huge variability in scales and shapes among species introduces a variety of challenges [26]. Motion capture in controlled settings with markers (providing ground truth skeleton and joint angle data for humans, horses and dogs [30, 60]), are not available for most animals. Generalised mesh surfaces may be used,

but often require multiple views and thousands of parameters, and do not guarantee consistency through time. In contrast, approximating an animal shape using a few-parameter morphable model can be both tractable and robust. Successful examples include swimmers [9, 43], birds [27, 58], mammals [2,6,28,39] and generic quadrupeds [7,67]. However, these methods expect opaque subjects with consistent textural appearances between views.

C. elegans has a simple geometric shape that can be well reconstructed from a midline skeleton and parametrised by curvature values along the body (see Sec. 3). This is the deformable template we look to fit to the data. Despite the apparent simplicity, each vertex of the discretised curve has two degrees of freedom (two curvature values) and as we use 128 vertices, our model is highly deformable and requires many parameters (although smoothness regularisation simplifies the problem somewhat). In contrast to deep-learning approaches, our model includes only a small number of explainable parameters and direct optimisation avoids lengthy training and dataset requirements.

C. elegans Numerous freely available software packages are capable of simultaneous tracking and skeletonising single or multiple worms in 2D using inexpensive microscopic imaging [5, 25, 44, 52, 53, 62] (see [24] for a review). Most of these skeletonisers combine image segmentation to separate the animal from the background with thinning of the mask to some midline pixels and fitting a spline.

The 3D reconstruction problem has received relatively little attention. Using at first two views [34] and then three, Kwon *et al*. [33] designed a motorised stage coupled with a real-time tracker to keep a worm in focus under high magnification in a 3D environment while capturing trajectories of up to 3 minutes. Thresholded images are lifted into 3D, intersected in voxel space and thinned [20] to produce a final skeleton. Kwon *et al*. omit camera modelling and assume perfectly parallel projections – assumptions that result in large errors for the data we use. Shaw *et al*. [49] employed light field microscopy to generate depth maps alongside images from a single viewpoint. A midline skeleton is generated by fitting a spline to the 3D coordinates of the central voxels. However, self-occlusions cannot be resolved and only relatively planar postures were investigated.

Salfelder *et al*. [45] and Yuval [63] both present 3D reconstruction algorithms using the three-camera set up and calibration described in [45]. In Salfelder *et al*. [45], a neural network is trained to identify 2D midlines from individual camera images before lifting into 3D voxel space. To account for changing camera parameters, a relative axial shift (dx, dy, dz) is optimised for each frame-triplet to maximise the voxel intersection before thinning. Remaining voxel coordinates are used as control points to fit a curve using a finite-element formulation. This approach works well when the midline is well detected in each of the views, but can fail on occluded postures or low-resolution, blurry images.

Yuval [63] uses a neural network to track head and tail points in 3D lab coordinates and a curve is fit between these fixed end points using a hill-climbing optimisation algorithm. Scoring is based on curve smoothness and pixel intensities at the projected curve points. This method works well when the head and tail are correctly identified but struggles, or requires manual correction, otherwise.

In our approach we find that incorporating the camera model parameters into the optimisation results in more robust and accurate results. This extends the idea proposed in Salfelder *et al*. [45] that adjusting the relative positions of the cameras could result in large gains in accuracy. It is likely that the relative shift adjustments, presented there, account for the changing optical properties.

3. Geometric model

Nematode shapes can be well approximated by a tapered cylinder and computed from a midline. We construct the midline curve in 3D using an object-centric parametrisation, separating shape from position and orientation to allow us to easily constrain and regularise the shape to stay within biologically-reasonable bounds. We discretise the curve into N equidistant vertices and encode the posture in curvature $K \in \mathbb{R}^{N\times 2}$ and length $l \in \mathbb{R}$ that fully define the shape up to a rigid-body transformation.

We express the 3D curve using the Bishop frame [8], given by TM^1M^2 where T is the normalised tangent of the curve and M^1, M^2 form an orthogonal basis along the midline. At vertex n, the curvature is $K_n = (m_n^1, m_n^2)$, where $m_n^1, m_n^2 \in \mathbb{R}$ are the curvature components along M^1, M^2. (The more familiar Frenet frame is less stable as it is undefined at zero-curvature points.) Numerical integration of a system of difference equations from starting point P_{init} and initial orientation $(T_{\text{init}}, M^1_{\text{init}}, M^2_{\text{init}})$ yields the curve path $P \in \mathbb{R}^{N\times 3}$. See supplementary material (SM) for details.

During optimisation, errors accumulate near the starting point, P_{init}, resulting in either parts of the curve moving faster than other or kinks developing (even with strong regularisation). To resolve this we sample an initial vertex index n_0 from a Gaussian distribution (subject to rounding) centred at the middle index at every optimisation step. Setting the starting point $P_{\text{init}} = P_{n_0}$ has the effect of continually shifting the discontinuity so kinks are never given the opportunity to develop (Fig. 2). Summarising the integration as F, the 3D curve is generated from the parameters:

$$(\hat{P}, \hat{T}, \hat{M}^1) = F\left(P_{n_0}, T_{n_0}, M^1_{n_0}, K, l, n_0\right). \qquad (1)$$

Each gradient update adjusts all curvature values K but the position and orientation only at the randomly selected n_0 vertex $(P_{n_0}, T_{n_0}, M^1_{n_0})$. Updating (P, T, M^1) at only

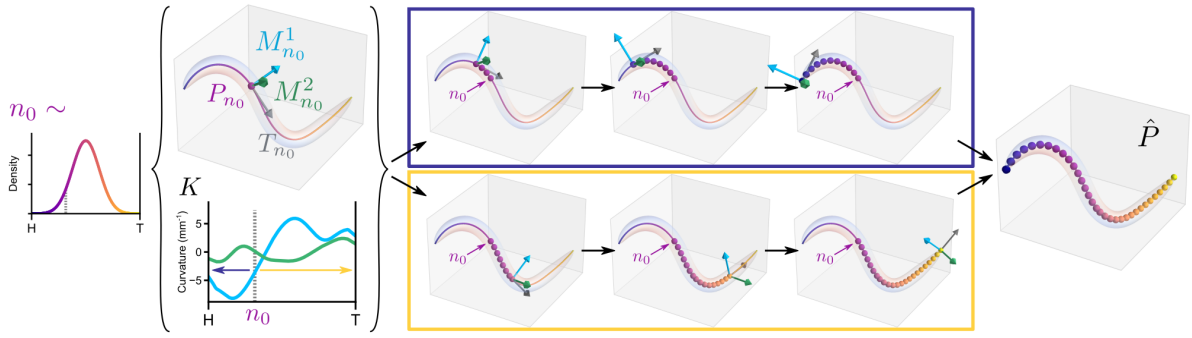

Figure 2. The 3D curve is traced out from initial point P_{n_0} and orientation frame $(T_{n_0}, M^1_{n_0}, M^2_{n_0})$. The index n_0 of the initial point is drawn from a normal distribution at each iteration to prevent kinks developing through repeated use of the same starting point. The final curve $\hat{P}$ is computed in two parts by integrating the Bishop equations with curvature K towards the head and tail separately.

this vertex produces a P that is inconsistent with the updated K. Therefore, after applying gradient updates we re-compute the full curve and orientation from n_0 and set (P, T, M^1) to the output $(\hat{P}, \hat{T}, \hat{M^1})$.

Since the curve describes a biological creature, we constrain the length l to $(l_{\text{min}}, l_{\text{max}})$ and limit the curvature by $|K_n| < 2\pi k_{\text{max}}$. The values of $(l_{\text{min}}, l_{\text{max}})$ we use vary depending on magnification but the bounds do not need to be tight and are in the range 0.5–$2\,\text{mm}$. The curvature constraint k_{max} is set by considering the number of circle achieved by a constant curvature curve and is fixed at 3.

4. Project, Render, Score

The core of the optimisation pipeline is separable into three main stages; project, render and score. The 3D curve $\hat{P}$ generated in Eq. (1) is *projected* through the camera models into 2D points that are *rendered* into images and then *scored* against the three views.

4.1. Project

The cameras are modelled using a triplet of pinhole camera models with tangential and radial distortion that project 3D points into image planes using perspective transformations. Each pinhole camera model offers a simple (15 parameters, $\{\eta_c\}$), tractable, approximation to the optical transformation. We also include relative shifts along the local coordinate axes, $\eta^s = (dx, dy, dz)$, shared between the three models, as proposed by Salfelder *et al.* [45]. Initial camera coefficients for the triplet-model are provided along with the recordings and typically give root mean squared reprojection errors up to 10 pixels ($\sim \mathcal{O}$(worm radius)).

Due to the initial calibration errors and changes in optical properties as the gelatin sets and is disturbed by the worms we re-calibrate the cameras at every frame by including the camera parameters in the optimisation step. To avoid an under-determined problem, after we have found a configuration that supports good reconstructions for a recording we fix all but the η^s parameters. Interestingly, we still see changes (up to $30\text{px} \sim 0.15\,\text{mm}$) in η^s but as this relates to the relative positioning it does not affect the posture reconstruction or long-term trajectories.

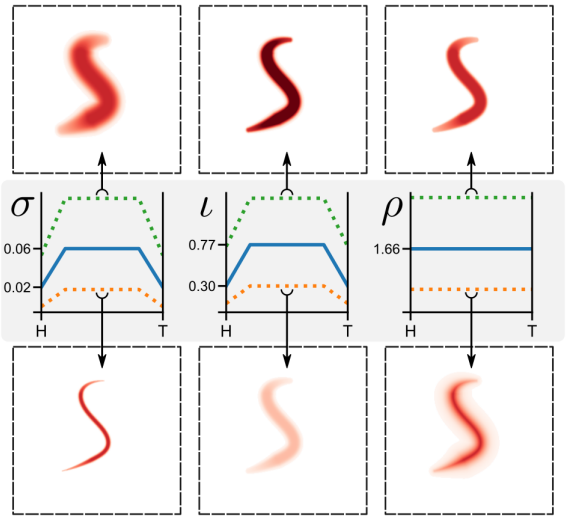

Figure 3. The rendering stage generates super-Gaussian blobs at each vertex position on the image. The shape of the blobs depends on the optimisable parameters: the scale σ, the intensity ι and the exponent used in the Gaussian ρ. σ and ι are tapered down to fixed minimum values at the head and tail. The effects of varying these parameters from a converged solution (blue curves) are shown above (green curves) and below (orange curves) each.

Projecting the 3D curve $\hat{P}$ through the camera-triplet model Γ with parameters $\eta = \{\eta_0, \eta_1, \eta_2, \eta^s\}$ generates 2D image points per view, which we combine as $Q = \Gamma(\hat{P}, \eta) \in \mathbb{R}^{3\times N\times 2}$.

4.2. Render

In order to evaluate the reconstruction directly against the raw data, we render the projected 2D midline points into

images using optimisable shape and rendering parameters. Since worm bodies are well approximated by tapered cylinders, in theory we only require maximum and minimum radius values and a tapering function. However, *C. elegans* are semi-transparent – increasingly so at the head and tail – and their internal anatomy has varying optical properties that diffract and distort the light. These challenges are further exacerbated by the worms often being out of focus in at least one of the views, therefore even an anatomically accurate model stands little chance of being correctly resolved.

We render realistic images by combining 2D super-Gaussian functions centred on each projected vertex. Crucially, we allow the rendering parameters to differ between cameras since the animal seldom has the same photometric qualities in different views. We optimise three parameters for each camera view c: $\sigma_c \in \mathbb{R}$ controls the spread, $\iota_c \in \mathbb{R}$ scales the intensity, and $\rho_c \in \mathbb{R}$ sharpens or softens the edges (Fig. 3). To capture the tapered shape we weight σ_c and ι_c from their optimisable values along the middle 60% to minimum values σ_{min} and ι_{min} at the ends and define the tapered outputs $\bar{\sigma}_c \in \mathbb{R}^N$ and $\bar{\iota}_c \in \mathbb{R}^N$ (SM). σ_{min} and ι_{min} are manually fixed for each recording to account for different magnification factors and worm size variability.

For each camera index c and vertex index n we define the rendered blob $B_{c,n} \in \mathbb{R}^{w \times w}$ (image size w) for pixel (i, j) as:

$$B_{c,n}(i,j) = \bar{\iota}_{c,n} \exp\left[-\left(\frac{(i - Q_{c,n,0})^2 + (j - Q_{c,n,1})^2}{2\bar{\sigma}_{c,n}^2}\right)^{\rho_c}\right]. \tag{2}$$

The stacks of blobs are combined to generate the complete renderings $R \in \mathbb{R}^{3 \times w \times w}$ by taking the maximum pixel value across all blobs: for pixel (i, j),

$$R_c(i,j) = \max\left\{B_{c,n}(i,j)\right\}_{n=0,\ldots,N-1}. \tag{3}$$

The orientation of the body directly affects the pixel intensity of both raw and rendered images. When pointing directly at a camera the peaks of the blobs cluster closely together and appear as a high-intensity (opaque) circle. Pointing laterally causes the peaks to spread out on the image revealing more of the lower-intensity tails. In both situations our blob-rendering approach approximates transparency effects in the raw images without the need to model complex intensity-orientation responses. Moreover, super-Gaussian blobs allow sharp outlines to be produced in one view by using a large exponent and flat-top blobs, and blurry images to be produced for another, using low intensity and high variance.

4.3. Score

In order to evaluate how well the curve represents the worm we require a way of distinguishing between worm-pixels and non-worm pixels such as dirt, bubbles, old tracks and even other worms. When the animal truly intersects with environmental interference it can be impossible to differentiate between the two, but in the majority of cases there exists a gap between the worm and the noise that is visible in at least one of the views. By ensuring that the curve corresponds to a single contiguous pixel mass in *all* of the images we are able to safely ignore other artefacts (Fig. 4).

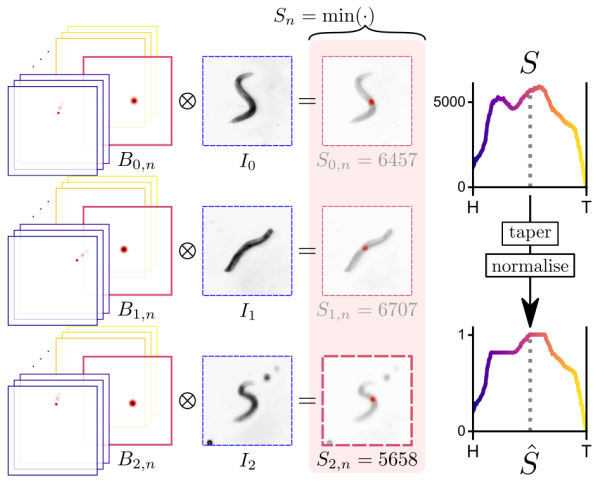

Figure 4. The 3D curve points are scored individually according to how well they match the three views. The triplet of blobs associated with vertex n ($B_{\cdot,n}$) are multiplied with the images I and summed. We take the minimum of the three sums and then taper these values from the midpoint-out.

To detect if the curve is bridging a gap, each vertex $\hat{P}_n$ is scored by correlating its corresponding blobs $B_{\cdot,n}$ (Sec. 4.2) with the images I. The raw score $S_n \in \mathbb{R}$ is defined:

$$S_n = \min\left\{\frac{\sum_{i,j} B_{c,n} \cdot I_c}{\bar{\sigma}_{c,n}\bar{\iota}_{c,n}}\right\}_{c=0,1,2} \tag{4}$$

where $\cdot$ is element-wise multiplication and the sum is taken over the image dimensions. By taking the minimum we ensure that vertices failing to match pixels in any one of the views will receive low scores regardless of how well they match pixels in the other views.

If the curve is bridging two disjoint groups of pixels that are visible in all three views this will present as two peaks in S. Since we are only interested in finding one object we restrict the scores to contain just one peak by tapering S from the middle-out to form the intermediate S'. Finally we normalise S' to get scores $\hat{S}$ relative to the peak:

$$S'_n = \begin{cases} \min\{S_n, S'_{n+1}\} & 0 \le n < N/2 \\ S_n & n = N/2 \\ \min\{S_n, S'_{n-1}\} & N/2 < n < N \end{cases} \tag{5}$$

$$\hat{S} = \frac{S'}{\max_n\{S'\}}. \tag{6}$$

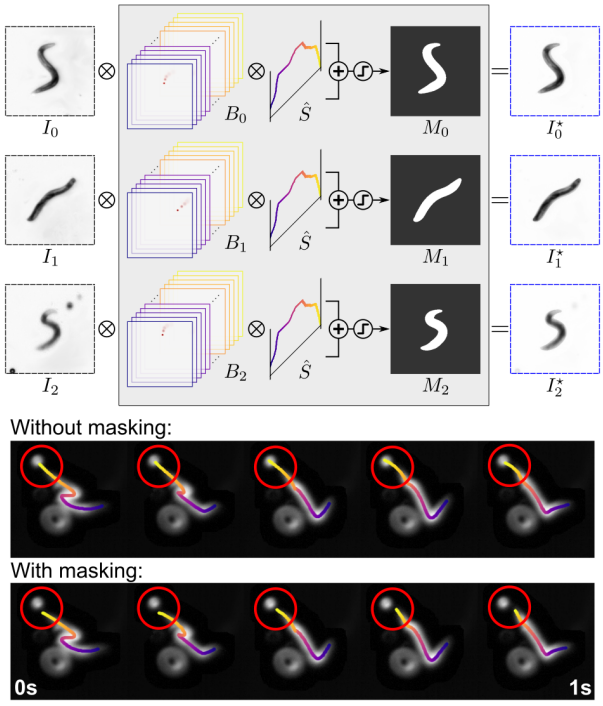

Figure 5. The noisy input images are cleaned by applying masks that force pixel-errors to be local to the current estimate. The blobs B are scaled by the relative scores $\hat{S}$, combined using the maximum pixel value across blobs and thresholded to form the masks M. The masks are applied to the raw input images I to generate the targets: $I^\star$. Masking ensures only a single contiguous pixel mass is detected. Without it, parts of the reconstruction can "stick" to nearby bubbles and other artefacts as shown below.

The final score profile $\hat{S}$ provides insight into how well the curve matches a contiguous pixel mass across all three views and how evenly that mass is distributed.

Masking From the score profile $\hat{S}$ we identify image areas that are more likely to contain the pixel masses that correspond to the worm. Masks $M \in \mathbb{R}^{3\times w\times w}$ applied to the input, $I^\star = M \cdot I$, focuses attention (and gradient) to only these areas of interest, consistently across all three views and exclude interference outside the masks (Fig. 5, see SM). Pixel intensities outside the masks are significantly reduced, but not zeroed in order to avoid stagnation in case the reconstruction completely misses the worm.

Centre-shifting The scores $\hat{S}$ also indicate the relative positioning of the curve over the target object. As the curve aligns with a pixel mass, vertices with high scores (apparently "converged") tend to lock into place thus hindering convergence of the rest of the object. For each frame, we use the previous frame solution as the starting point, so the majority of points rapidly converge. However, errors introduced at the tips remain as they are insufficient to generate the collective shift required. The effect can easily be identified from an unbalanced score profile (Fig. 6) and rectified by periodically shifting the curve along its length between gradient descent optimisation steps (see SM).

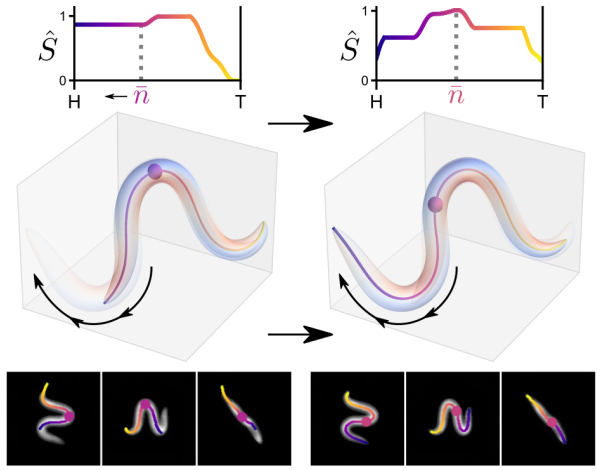

Figure 6. As the animal moves along the path of its midline the tail may be left behind (left column). This can be identified from an unbalanced score profile $\hat{S}$. By periodically shifting the curve along its length (adding new curvature values at one end and discarding from the other) the centroid index ($\bar{n}$) of the scores can be centred. Gradient descent optimisation then updates the new curvature values so the curve matches the target (right column).

5. Optimisation

The main pixel-loss to be minimised is defined as:

$$\mathcal{L}_{\text{px}} = \frac{1}{3w^2}\sum_{c,i,j}(R_c(i,j) - I_c^\star(i,j))^2. \tag{7}$$

To improve head and tail detection we also minimise a scores-loss,

$$\mathcal{L}_{\text{sc}} = \frac{\max(S')N}{\sum_n S''_n}, \text{ where} \tag{8}$$

$$S''_n = S'_n\left(\frac{2n-(N-1)}{N-1}\right)^2, \tag{9}$$

that is quadratically weighted towards the tips where the scores are naturally lower due to the transparency.

In addition we include a number of regularisation terms. To keep the curve smooth we define

$$\mathcal{L}_{\text{sm}} = \sum_{n=1}^{N-1}|K_n - K_{n-1}|^2, \tag{10}$$

where $|\cdot|$ is the l^2-norm. To ensure all parameters change

smoothly between frames we set

$$\mathcal{L}_{\text{t}} = \sum_{x \in \{l, K, \hat{P}, \eta, \sigma, \iota, \rho\}} |x^{\text{prev}} - x|^2, \tag{11}$$

where x^{prev} refers to the frozen value of the variable from the previous frame. And to avoid self-intersections, we use

$$d_{n,m} = |\hat{P}_n - \hat{P}_m|, \tag{12}$$

$$d'_{n,m} = \frac{1}{3}\sum_c \bar{\sigma}_{c,n} + \frac{1}{3}\sum_c \bar{\sigma}_{c,m}, \text{ and} \tag{13}$$

$$\mathcal{L}_{\text{i}} = \sum_{n=0}^{N - N/k_{\max} - 1} \sum_{m=n+N/k_{\max}}^{N-1} \begin{cases} \frac{d'_{n,m}}{d_{n,m}}, & \text{if } d_{n,m} < d'_{n,m} \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

A loss is incurred, $\mathcal{L}_{\text{i}} > 0$, when two points which are sufficiently far apart ($> N/k_{\max}$) along the curve come within a distance defined by the sum of their mean rendering variances (since these approximate the worm's radius). Eq. (14) forces the algorithm to find postures that are always feasible even during self-occlusions and complex manoeuvres.

The losses are combined in a weighted sum to yield the final optimisation target:

$$\mathcal{L} = \omega_{\text{px}}\mathcal{L}_{\text{px}} + \omega_{\text{sc}}\mathcal{L}_{\text{sc}} + \omega_{\text{sm}}\mathcal{L}_{\text{sm}} + \omega_{\text{t}}\mathcal{L}_{\text{t}} + \omega_{\text{i}}\mathcal{L}_{\text{i}}. \tag{15}$$

Values of ω used in our experiments are included in the SM.

To achieve robust reconstructions it is important that the curve parameters learn fastest, then the rendering parameters and finally the camera parameters. Imposing this hierarchy of rates ensures camera model stability and prevents the renderer from over-blurring the edges (as it tries to "reach" the pixels). Thus, movement between frames is primarily captured through curve deformations. We use learning rates $\lambda_p = 1\text{e}{-3}$ for the curve parameters $\{P, T, M^1, K, l\}$, $\lambda_r = 1\text{e}{-4}$ for the rendering parameters $\{\sigma, \iota, \rho\}$ and $\lambda_\eta = 1\text{e}{-5}$ for the camera parameters η.

The curve is initialised as a small ($\sim 0.2\,\text{mm}$), randomly oriented straight line centred in the field of view of all three cameras. We slowly increase the length to $l_{\min}$ over the first 200-500 steps as the curve gets positioned and orientated.

The pipeline is constructed using PyTorch [64] and the loss minimised is using Adam [31] with periodic centre-shifting of the curve vertices. Learning rates are decreased by a factor of 0.8 for every 5 steps taken without improvement in $\mathcal{L}$ to a minimum of $1\text{e}{-6}$ until convergence is detected. Subsequent frames are instantiated with the solution from the previous frame for efficiency and to maintain consistency through complex sequences of self-occluding postures. Example videos showing the effects of varying some of the options on the optimisation are described in SM.

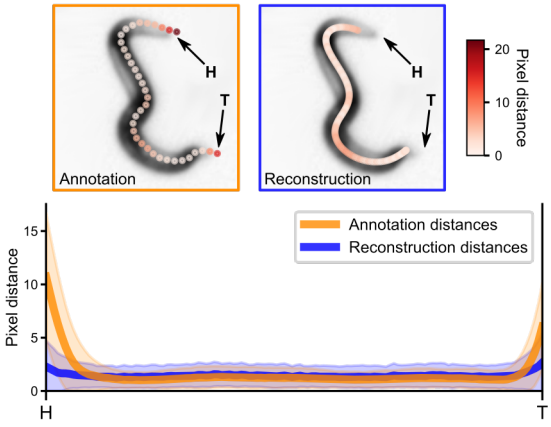

Figure 7. Validation against 487 manual annotations. At the top we show an example of an annotated frame (left, orange) alongside a projection of our matching 3D midline (right, blue). Below we plot the sample averages $\pm$2std. We find our midlines are consistently close to annotated points (blue curve), but annotations typically extend further into the head and tail regions (orange curve).

6. Results

Using our method we generate high quality 3D midline reconstructions for 43 of 44 recordings. One fails due to excessive coiling of the worm. Significant occlusions also occur during successful reconstructions and when combined with loss of focus can cause the shape to be lost. Video clips of good and poor reconstructions through challenging environmental conditions are described in SM along with ablation results to show benefits of each component.

We compare 2D reprojections of our midlines against 487 manual annotations that were produced from single images in isolation and contain a varying number of unordered points. We calculate the minimum distance from each annotated point to any reconstructed point and vice-versa and find that our midlines consistently come close ($\sim$2px) to hand-annotated points (Fig. 7). Annotated points at the ends show an increased distance ($\sim$10px) to our midline points. This shows that our curves generally fall short of reaching the very tips of the worm by $\sim \mathcal{O}(\text{worm radius})$.

Our method significantly outperforms previous methods developed using the same dataset [45, 63] when evaluated against the manual annotations (SM), but these only cover a selection of hand-picked examples. For a large-scale comparison we take 3D midlines and camera parameters found by each method and, using our pipeline, render them to generate comparable images (re-optimising the render parameters for their midlines, see SM). We skip the scoring and masking and calculate $\mathcal{L}_{\text{px}}$. The results (Fig. 8) show our method consistently produces shapes that more closely match the raw images. The biggest advantage over previous approaches is the improvement in robustness; we recover

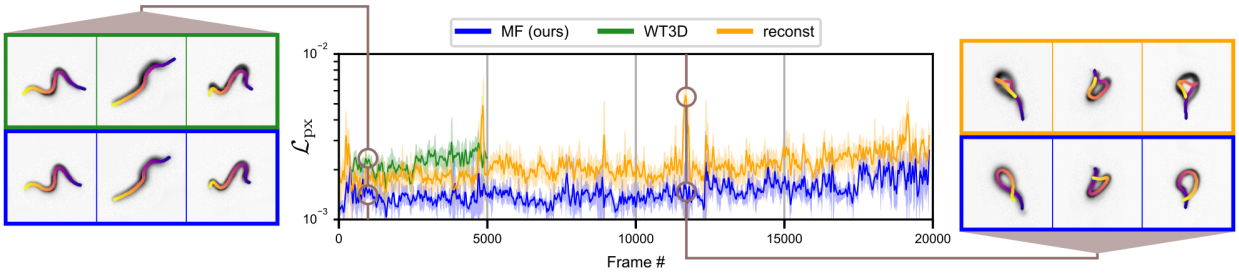

Figure 8. A comparison between our Midline Finder (MF), Yuval's Worm-Tracker 3D (WT3D) [63] and Salfelder *et al.*'s 'reconst' [45] methods across a single trial ($\sim$13 min). In the majority of cases our method generates midlines that better match the data (lower pixel losses, $\mathcal{L}_{\text{px}}$). We show moving averages over 25 frames ($\sim$1 s) with shaded areas indicating $\pm$2std.

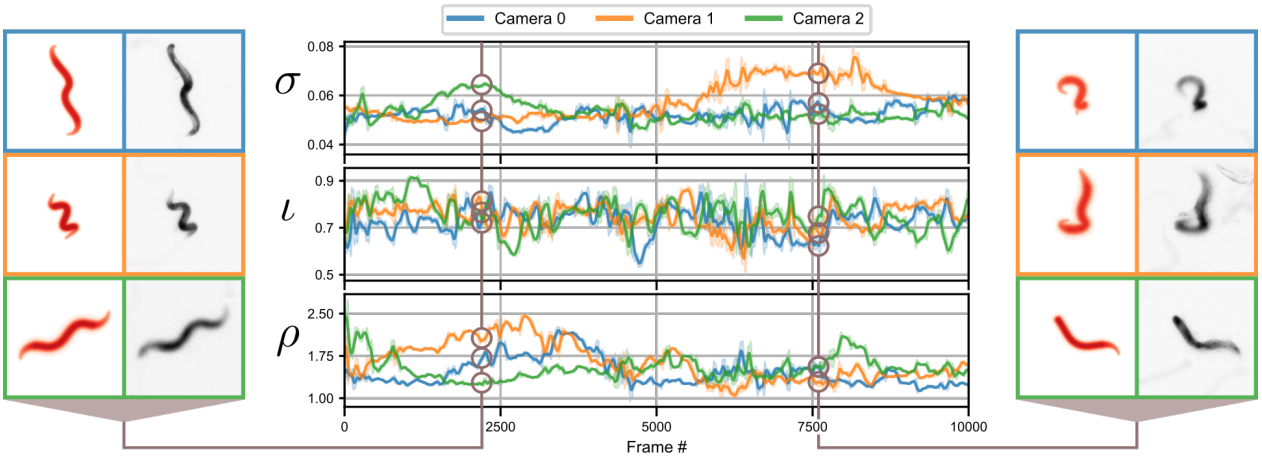

Figure 9. The rendering parameters change continually over the course of a recording to capture optical changes. Clear images (*e.g.* early frames in cameras 0 and 1, switching to late frames in camera 2) are consistent with small values of σ and large values of ρ. Blurry images (early camera 2, late camera 1) use high σ and small ρ. We show moving averages over 25 frames ($\sim$1 s) with shaded areas indicating $\pm$2std. Example comparisons between the renders (red) and raw images (grey) are shown on either side.

4 h 37 min (ours) versus 1 h 32 min [45] and 45 min [63].

Fig. 9 shows the rendering parameters during a trial as the worm moves in and out of focus in the different cameras. Clearer images result in smaller values of σ and larger values of ρ. The fluctuations in intensity ι are due in part to the posture of the worm in relation to the camera; when it is pointing directly towards the camera we see higher values of ι used to capture the darker image observed and when the shape is perpendicular to the camera we see lower values of ι to emulate the worm's transparency. All three parameters work in tandem to produce the final effect.

7. Conclusion

We present a robust and reliable framework for the 3D reconstruction of a microscopic, semi-transparent subject moving through a fluid and evaluate against two other algorithms and manually annotations. The key contribution of our approach – constructing unique differentiable renderings for each view – allows us to solve shape reconstruction and camera parameter optimisation by direct image comparison. This avoids feature extraction and correspondence matching, and hence offers a powerful alternative when those approaches are not well-suited, *e.g.* due to the variation in appearance between views.

Multi-view microscopic camera calibration, imaging through fluids and parametric model fitting of semi-transparent subjects are challenges that have received little attention in the literature. While we have focused here on constructing a curve to fit a microscopic worm from three views, our method could be applied to the 3D reconstruction of arbitrary shape models at any scale using any number of viewpoints. Rendering points with adaptable super-Gaussian functions presents an effective solution to transparency and focal issues, but more generally, our results indicate that our direct optimisation approach may offer an effective alternative to contemporary methods for 3D approximation of generic objects from a limited number of silhouette-like images.

References

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2D human pose estimation: New benchmark and state of the art analysis. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3686–3693. IEEE, June 2014. 2

[2] Praneet C. Bala, Benjamin R. Eisenreich, Seng Bum Michael Yoo, Benjamin Y. Hayden, Hyun Soo Park, and Jan Zimmermann. Automated markerless pose estimation in freely moving macaques with OpenMonkeyStudio. *Nat Commun*, 11(1):4560, Sept. 2020. 3

[3] Jerrold L Belant, Joshua J Millspaugh, James A Martin, and Robert A Gitzen. Multi-dimensional space use: The final frontier. *Front. Ecol. Environ.*, 10(1):11–12, Feb. 2012. 1

[4] Florian Berlinger, Melvin Gauci, and Radhika Nagpal. Implicit coordination for 3D underwater collective behaviors in a fish-inspired robot swarm. *Sci. Robot.*, 6(50):eabd8668, Jan. 2021. 1

[5] Stefano Berri, Jordan H. Boyle, Manlio Tassieri, Ian A. Hope, and Netta Cohen. Forward locomotion of the nematode *C. elegans* is achieved through modulation of a single gait. *Hfsp J.*, 3(3):186–193, June 2009. 3

[6] Benjamin Biggs, Oliver Boyne, James Charles, Andrew Fitzgibbon, and Roberto Cipolla. Who left the dogs out? 3d animal reconstruction with expectation maximization in the loop. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 195–211. Springer, 2020. 3

[7] Benjamin Biggs, Thomas Roddick, Andrew Fitzgibbon, and Roberto Cipolla. Creatures great and smal: Recovering the shape and motion of animals from video. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part V 14*, pages 3–19. Springer, 2019. 3

[8] Richard L. Bishop. There is more than one way to frame a curve. *Amer. Math. Monthly*, 82(3):246–251, Mar. 1975. 3

[9] Thomas J. Cashman and Andrew W. Fitzgibbon. What shape are dolphins? building 3D morphable models from 2D images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(1):232–244, Jan. 2013. 3

[10] Ching-Hang Chen and Deva Ramanan. 3D human pose estimation = 2D pose estimation + matching. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7035–7043. IEEE, July 2017. 2

[11] Nathan W. Cooper, Thomas W. Sherry, and Peter P. Marra. Modeling three-dimensional space use and overlap in birds. *Auk*, 131(4):681–693, Oct. 2014. 1

[12] Amael Delaunoy and Marc Pollefeys. Photometric bundle adjustment for dense multi-view 3D modeling. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1486–1493. IEEE, June 2014. 2

[13] Olivier Faugeras and Quang-Tuan Luong. *The Geometry of Multiple Images*. The MIT Press, 2001. 2

[14] Alessandro Ferrarini, Giuseppe Giglio, Stefania Caterina Pellegrino, Anna Grazia Frassanito, and Marco Gustin. A new methodology for computing birds' 3D home ranges. *Avian Res*, 9(1):1–6, May 2018. 1

[15] Lise Frézal and Marie-Anne Félix. The natural history of model organisms: *C. elegans* outside the petri dish. *eLife*, 4:e05849, Mar. 2015. 2

[16] Kui Fu, Jiansheng Peng, Qiwen He, and Hanxiao Zhang. Single image 3D object reconstruction based on deep learning: A review. *Multimed Tools Appl*, 80(1):463–498, Sept. 2020. 2

[17] Marie-Anne Félix and Christian Braendle. The natural history of caenorhabditis elegans. *Curr. Biol.*, 20(22):R965–R969, Nov. 2010. 2

[18] P. Georgel, S. Benhimane, and N. Navab. A unified approach combining photometric and geometric information for pose estimation. In *Procedings of the British Machine Vision Conference 2008*, pages 1–10. Citeseer, British Machine Vision Association, 2008. 2

[19] Riza Alp Guler, Natalia Neverova, and Iasonas Kokkinos. DensePose: Dense human pose estimation in the wild. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7297–7306. IEEE, June 2018. 2

[20] Zicheng Guo and Richard W. Hall. Parallel thinning with two-subiteration algorithms. *Commun. ACM*, 32(3):359–373, Mar. 1989. 3

[21] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Mar. 2004. 2

[22] Robert I. Holbrook and Theresa Burt de Perera. Three-dimensional spatial cognition: Information in the vertical dimension overrides information from the horizontal. *Anim Cogn*, 14(4):613–619, Mar. 2011. 1

[23] C.T. Huang and O.R. Mitchell. Dynamic camera calibration. In *Proceedings of International Symposium on Computer Vision - ISCV*, pages 169–174. IEEE, IEEE Comput. Soc. Press, 1995. 2

[24] Steven J. Husson, Wagner S. Costa, Cornelia Schmitt, and Alexander Gottschalk. Keeping track of worm trackers. *WormBook*, pages 1–17, Sept. 2012. 3

[25] Avelino Javer, Michael Currie, Chee Wai Lee, Jim Hokanson, Kezhi Li, Céline N. Martineau, Eviatar Yemini, Laura J. Grundy, Chris Li, QueeLim Ch'ng, William R. Schafer, Ellen A. A. Nollen, Rex Kerr, and André E. X. Brown. An open-source platform for analyzing and sharing worm-behavior data. *Nat Methods*, 15(9):645–646, Aug. 2018. 3

[26] Le Jiang, Caleb Lee, Divyang Teotia, and Sarah Ostadabbas. Animal pose estimation: A closer look at the state-of-the-art, existing gaps and opportunities. *Comput. Vis. Image Und.*, 222:103483, Sept. 2022. 1, 2

[27] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7122–7131. IEEE, June 2018. 2, 3

[28] Angjoo Kanazawa, Shahar Kovalsky, Ronen Basri, and David Jacobs. Learning 3d deformation of animals from 2d images. In *Computer Graphics Forum*, volume 35, pages 365–374. Wiley Online Library, 2016. 3

[29] Isinsu Katircioglu, Bugra Tekin, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua. Learning latent representations of

3D human pose with deep neural networks. *Int J Comput Vis*, 126(12):1326–1341, Jan. 2018. 2

[30] Sinead Kearney, Wenbin Li, Martin Parsons, Kwang In Kim, and Darren Cosker. RGBD-dog: Predicting canine pose from RGBD sensors. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8336–8345. IEEE, June 2020. 1, 2

[31] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[32] Nikos Kolotouros, Georgios Pavlakos, Michael Black, and Kostas Daniilidis. Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2252–2261. IEEE, Oct. 2019. 2

[33] Namseop Kwon, Ara B. Hwang, Young-Jai You, Seung-Jae V. Lee, and Jung Ho Je. Dissection of *C. elegans* behavioral genetics in 3-d environments. *Sci Rep*, 5(1):1–9, May 2015. 3

[34] Namseop Kwon, Jaeyeon Pyo, Seung-Jae Lee, and Jung Ho Je. 3-d worm tracker for freely moving *C. elegans*. *PLoS ONE*, 8(2):e57484, Feb. 2013. 3

[35] Wu Liu, Qian Bao, Yu Sun, and Tao Mei. Recent advances of monocular 2D and 3D human pose estimation: A deep learning perspective. *ACM Comput. Surv.*, 55(4):1–41, Nov. 2022. 2

[36] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(5828):133–135, Sept. 1981. 2

[37] Simone Macrì, Daniele Neri, Tommaso Ruberto, Violet Mwaffo, Sachit Butail, and Maurizio Porfiri. Three-dimensional scoring of zebrafish behavior unveils biological phenomena hidden by two-dimensional analyses. *Sci Rep*, 7(1):1–10, May 2017. 1

[38] Julieta Martinez, Rayat Hossain, Javier Romero, and James J. Little. A simple yet effective baseline for 3d human pose estimation. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2640–2649. IEEE, Oct. 2017. 2

[39] Valsamis Ntouskos, Marta Sanzari, Bruno Cafaro, Federico Nardi, Fabrizio Natola, Fiora Pirri, and Manuel Ruiz. Component-wise modeling of articulated objects. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2327–2335. IEEE, Dec. 2015. 3

[40] Onur Özyeşil, Vladislav Voroninski, Ronen Basri, and Amit Singer. A survey of structure from motion. *Acta Numer.*, 26:305–364, May 2017. 2

[41] Brian L. Partridge, Tony Pitcher, J. Michael Cullen, and John Wilson. The three-dimensional structure of fish schools. *Behav Ecol Sociobiol*, 6(4):277–288, Mar. 1980. 1

[42] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G. Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3D human pose. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7025–7034. IEEE, July 2017. 2

[43] Mukta Prasad, Andrew Fitzgibbon, Andrew Zisserman, and Luc Van Gool. Finding nemo: Deformable object class modelling using curve matching. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1720–1727. IEEE, IEEE, June 2010. 3

[44] Daniel Ramot, Brandon E. Johnson, Tommie L. Berry, Lucinda Carnell, and Miriam B. Goodman. The parallel worm tracker: A platform for measuring average speed and drug-induced paralysis in nematodes. *PLoS ONE*, 3(5):e2208, May 2008. 3

[45] Felix Salfelder, Omer Yuval, Thomas P Ilett, David C Hogg, Thomas Ranner, and Netta Cohen. Markerless 3D spatio-temporal reconstruction of microscopic swimmers from video. In *Visual observation and analysis of Vertebrate And Insect Behavior 2020*, 2021. 2, 3, 4, 7, 8

[46] Hinrich Schulenburg and Marie-Anne Félix. The natural biotic environment of *Caenorhabditis elegans*. *Genetics*, 206(1):55–86, May 2017. 2

[47] S.M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR'06)*, volume 1, pages 519–528. IEEE, IEEE, 2006. 2

[48] William Irvin Sellers and Eishi Hirasaki. Markerless 3D motion capture for animal locomotion studies. *Biology Open*, 3(7):656–668, June 2014. 2

[49] Michael Shaw, Haoyun Zhan, Muna Elmi, Vijay Pawar, Clara Essmann, and Mandayam A. Srinivasan. Three-dimensional behavioural phenotyping of freely moving *C. elegans* using quantitative light field microscopy. *PLoS ONE*, 13(7):e0200108, July 2018. 3

[50] Colin A. Simpfendorfer, Esben M. Olsen, Michelle R. Heupel, and Even Moland. Three-dimensional kernel utilization distributions improve estimates of space use in aquatic animals. *Can. J. Fish. Aquat. Sci.*, 69(3):565–572, Mar. 2012. 1

[51] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5693–5703. IEEE, June 2019. 2

[52] Nicholas A Swierczek, Andrew C Giles, Catharine H Rankin, and Rex A Kerr. High-throughput behavioral analysis in *C. elegans*. *Nat Methods*, 8(7):592–598, June 2011. 3

[53] Raphael Sznitman, Manaswi Gupta, Gregory D. Hager, Paulo E. Arratia, and Josué Sznitman. Multi-environment model estimation for motility analysis of caenorhabditis elegans. *PLoS ONE*, 5(7):e11631, July 2010. 3

[54] Diane Theriault, Zheng Wu, Nickolay I Hristov, Sharon M Swartz, Kenneth S Breuer, Thomas H Kunz, and Margrit Betke. Reconstruction and analysis of 3D trajectories of Brazilian free-tailed bats in flight. In *20th Int. Conf. on Pattern Recognition*, pages 1–4, 2010. 1

[55] Alexander Toshev and Christian Szegedy. DeepPose: Human pose estimation via deep neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1653–1660. IEEE, June 2014. 2

[56] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle adjustment — a modern synthesis. In *Vision Algorithms: Theory and Practice*, pages 298–372. Springer Berlin Heidelberg, 2000. 2

[57] R. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE J. Robot. Automat.*, 3(4):323–344, Aug. 1987. 2

[58] Sara Vicente and Lourdes Agapito. Balloon shapes: Reconstructing and deforming objects with volume from images. In *2013 International Conference on 3D Vision*, pages 223–230. IEEE, IEEE, June 2013. 3

[59] J. Weng, P. Cohen, and M. Herniou. Camera calibration with distortion models and accuracy evaluation. *IEEE Trans. Pattern Anal. Machine Intell.*, 14(10):965–980, 1992. 2

[60] Nils Wilhelm, Anna Vögele, Rebeka Zsoldos, Theresia Licka, Björn Krüger, and Jürgen Bernard. FuryExplorer: Visual-interactive exploration of horse motion capture data. In *SPIE Proceedings*, volume 9397, pages 148–162. SPIE, SPIE, Feb. 2015. 1, 2

[61] Shangzhe Wu, Christian Rupprecht, and Andrea Vedaldi. Unsupervised learning of probably symmetric deformable 3D objects from images in the wild. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–10. IEEE, June 2020. 2

[62] Eviatar Yemini, Rex A. Kerr, and William R. Schafer. Tracking movement behavior of multiple worms on food. *Cold Spring Harb Protoc*, 2011(12):pdb.prot067025, Dec. 2011. 3

[63] Omer Yuval. *The neuromechanical control of Caenorhabditis elegans head motor behaviour in 3D environments*. PhD thesis, University of Leeds, 2022. 3, 7, 8

[64] Sergey Zagoruyko, Adam Lerer, Tsung-Yi Lin, PedroO. Pinheiro, Sam Gross, Soumith Chintala, and Piotr Dollar. A MultiPath network for object detection. In *Procedings of the British Machine Vision Conference 2016*. British Machine Vision Association, 2016. 7

[65] Liqun Zhu and Wei Weng. Catadioptric stereo-vision system for the real-time monitoring of 3D behavior in aquatic animals. *Physiology & Behavior*, 91(1):106–119, May 2007. 1

[66] Michael Zollhöfer, Patrick Stotko, Andreas Görlitz, Christian Theobalt, Matthias Nießner, Reinhard Klein, and Andreas Kolb. State of the art on 3D reconstruction with RGB-d cameras. In *Computer graphics forum*, volume 37, pages 625–652. Wiley Online Library, 2018. 2

[67] Silvia Zuffi, Angjoo Kanazawa, David W. Jacobs, and Michael J. Black. 3D menagerie: Modeling the 3D shape and pose of animals. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6365–6373. IEEE, July 2017. 3