LiftPose3D, a deep learning-based approach for transforming 2D to 3D pose in laboratory animals

Adam Gosztolai*†1, Semih Günel*†1,2, Marco Pietro Abrate[1], Daniel Morales[1], Victor Lobato Ríos[1], Helge Rhodin[3], Pascal Fua[2], and Pavan Ramdya*1

[1]Neuroengineering Laboratory, Brain Mind Institute & Interfaculty Institute of Bioengineering, EPFL, Lausanne, Switzerland
[2]Computer Vision Laboratory, EPFL, Lausanne, Switzerland
[3]Department of Computer Science, UBC, Vancouver, Canada

Abstract

Markerless 3D pose estimation has become an indispensable tool for kinematic studies of laboratory animals. Most current methods recover 3D pose by multi-view triangulation of deep network-based 2D pose estimates. However, triangulation requires multiple, synchronised cameras per keypoint and elaborate calibration protocols that hinder its widespread adoption in laboratory studies. Here, we describe LiftPose3D, a deep network-based method that overcomes these barriers by reconstructing 3D poses from a single 2D camera view. We illustrate LiftPose3D's versatility by applying it to multiple experimental systems using flies, mice, and macaque monkeys and in circumstances where 3D triangulation is impractical or impossible. Thus, LiftPose3D permits high-quality 3D pose estimation in the absence of complex camera arrays, tedious calibration procedures, and despite occluded keypoints in freely behaving animals.

1 Introduction

To identify the principles underlying how actions arise from neural circuit dynamics, one must first be able to make precise measurements of behavior in laboratory experiments. Paired with new methods for recording neuronal populations in behaving animals [1, 2, 3, 4], recent innovations in 3-dimensional (3D) pose estimation—tracking body parts of interest—promise to accelerate the discovery of these neural control principles. 3D pose estimation is typically accomplished by triangulating 2-dimensional (2D) poses acquired using multiple camera views and deep network-based markerless keypoint tracking algorithms [5, 6, 7, 8, 9, 10, 11, 12, 13]. Notably, triangulation requires that every tracked keypoint be visible from at least two synchronized cameras [14] and that each camera is first calibrated by hand [15, 16] or, as in DeepFly3D, by solving a non-convex optimization problem [7]. These expectations are expensive and often difficult to meet, particularly in space-constrained experimental systems that also house sensory stimulation devices [1, 2, 17]. Additionally, in freely behaving animals, limb keypoints are often hidden in some camera views due to self-occlusions, making 3D triangulation impossible for those keypoints.

Due to the challenges associated with 3D pose estimation, most studies have favored the simplicity and higher throughput afforded by 2D pose estimation using one camera and a single viewpoint [18, 19, 6, 20, 10, 5]. However, projected 2D poses can result from multiple distinct 3D poses, and thus true 3D joint configurations remain unknown. Computer vision research on human pose estimation has long attempted to recover 3D poses by regressing 3D pose ground-truth from 2D pose datasets [21, 22, 23, 24], with more recent deep learning-based methods achieving very high accuracy [25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37]. However, until now, none of these powerful techniques have been adapted to satisfy the unique challenges of laboratory animal studies, including the absence of large training datasets.

*corresponding authors: adam.gosztolai@epfl.ch; semih.gunel@epfl.ch; pavan.ramdya@epfl.ch
†equal contribution

Here, we introduce LiftPose3D, a deep learning-based tool for rapid and reliable frame-by-frame 3D pose estimation of tethered and freely behaving laboratory animals. Our method is based on a neural network that has been used to map or lift 2D human poses into 3D poses using only a single camera view [33]. This specific architecture was chosen for its simplicity: because it does not require temporal information, or a skeletal graph, it is more easily generalizable. We demonstrate data transformations and adaptations to network training which enable accurate 3D pose estimation given only small amounts of training data (10^3 - 10^4 images) and across a wide range of animals and experimental systems. For example, we show that alignment by registration allows the network to learn the bilateral symmetries inherent in animal poses. This permits 3D pose estimation despite self-occlusions in freely behaving animals and weakens the effect of outliers in training data. Additionally, we find that pretrained LiftPose3D networks can be adapted and generalized to different experimental domains by matching pose statistics across datasets and by coarse-graining the network across image resolutions.

We illustrate these findings in several experimental scenarios. First, we apply LiftPose3D to data from tethered adult *Drosophila* [7] and freely behaving macaque monkeys [8] to show how it can dramatically reduce the number of cameras required for 3D pose estimation. We make these pretrained networks and our code publicly available to be applied to new laboratory experiments. Second, we demonstrate lifting despite self-occlusions in freely behaving *Drosophila* and mice [38]. Third, we apply pretrained LiftPose3D networks to predict realistic 3D poses from completely different experimental systems including from a previously published dataset consisting of a single viewpoint behavioral video [19]. Thus, we can effectively resurrect old data for new kinds of kinematic analyses.

2 Results

Given that a keypoint j of interest is visible from at least two cameras c, with corresponding 2D coordinates $\mathbf{x}_{c,j}$, then its 3D coordinates $\mathbf{X}_j$ in a global world reference frame can be obtained by triangulation. Here we use triangulated 3D positions as benchmark ground truths data against which to assess LiftPose3D, a method that focuses on cases in which some joints are only visible from a single view either because there is only one camera, or due to self-occlusions in some views. Rather than considering keypoints independently, our goal is to simultaneously predict the coordinates of n keypoints $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_n)$—the 3D pose—in a global world reference frame from their respective 2D coordinates $\mathbf{x}_c = (\mathbf{x}_{c,1}, \ldots, \mathbf{x}_{c,n})$ viewed from any of N fixed cameras $c = 1, \ldots, N$ (e.g., see **Figure 1**A, illustrating six fixed cameras).

The basis of LiftPose3D is to estimate 3D pose X by learning a nonlinear mapping between a library of ground truth 2D poses viewed from any camera c and their corresponding 3D poses. Formally, this operation is encoded in a *lifting* function f mapping all keypoints visible from any camera c to their corresponding camera-centered coordinates $\mathbf{Y}_c = f(\mathbf{x}_c)$. These 3D points are then converted to world coordinates $\mathbf{X} = \phi_c^{-1}(\mathbf{Y}_c)$ by inverting the affine transformation ϕ_c mapping from world to camera coordinates. The lifting function f can in turn be approximated by a deep neural network $F(\mathbf{x}_c; \Theta)$ where Θ represents the network weights controlling the behavior of F. We use the network architecture from [33], composed of fully connected layers, batch-norm, and dropout layers connected with skip connections (**Figure 1C,D**). To obtain f for a specific application, the parameters Θ are trained by minimizing the discrepancy between lifted and triangulated poses over all cameras c and keypoints j, meaning

$$\mathcal{J}_1(\Theta) := \sum_{c=1}^{N} \sum_{j=1}^{n} ||(F(\mathbf{x}_c;\, \Theta))_j - \mathbf{Y}_{c,j}||_2^2 \,. \tag{1}$$

Previously, this network had been trained on 3.6×10^6 fully annotated 2D-3D human pose pairs for many different human behaviors. Here we co-opted this network architecture to address the unique challenges of lifting 3D poses in animal experiments. Specifically, our developments permit this network to be used in the laboratory to now (i) work with a vastly smaller amount of training data (between 10^3-10^4 pose pairs), (ii) reduce the number of cameras needed to predict full 3D poses, (iii) train with only partially annotated ground truth 3D poses due to self-occlusions, and (iv) generalize a single pretrained network across experimental systems and domains by data augmentation. These algorithmic contributions make it possible to perform 3D pose estimation in a number of important, previously inaccessible experimental scenarios which we illustrate below.

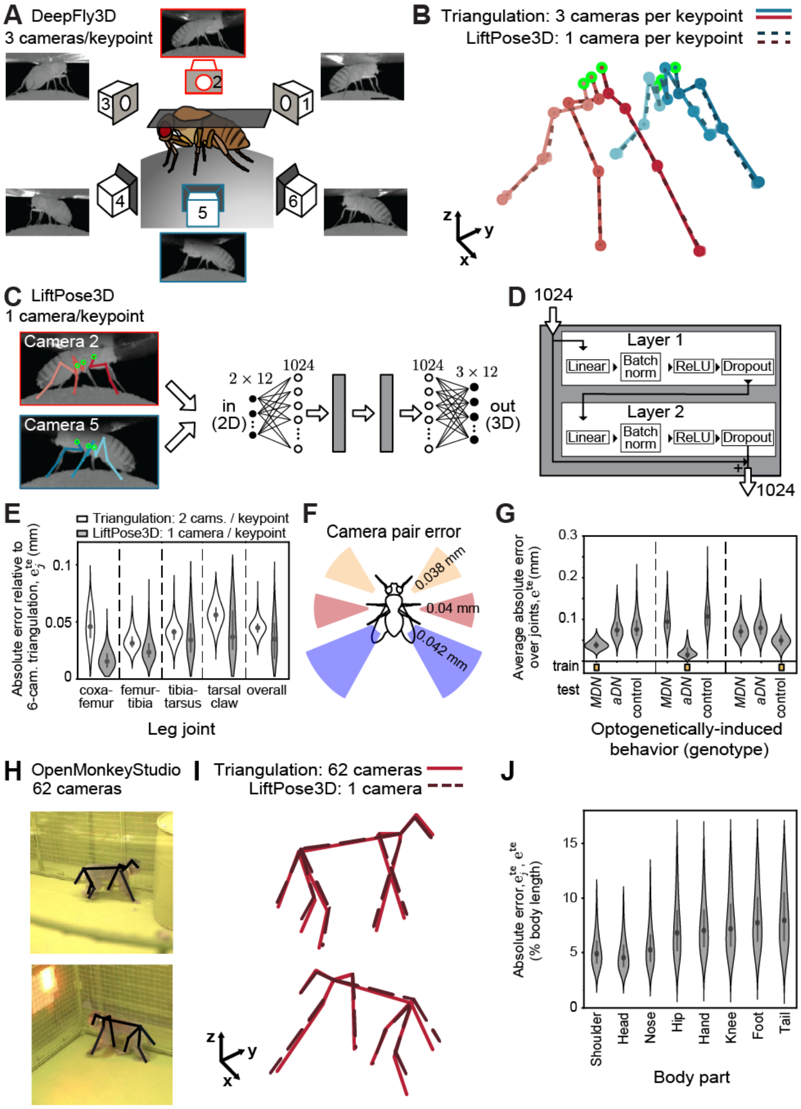

Figure 1: **LiftPose3D reduces the number of cameras needed for full 3D pose estimation. A** 3D poses of tethered *Drosophila* can be triangulated using six camera views (3 cameras per keypoint). LiftPose3D can predict accurate 3D poses using only one camera per side (red and blue). **B** 3D joint positions for right (red) and left (blue) legs of a tethered fly obtained using triangulation with three camera views per keypoint (solid lines), or using LiftPose3D with one camera view per side (dashed lines). Thorax-coxa root joints are highlighted (green). **C** As inputs, LiftPose3D takes deep network-derived half-body 2D poses for 15 joints per camera (red and blue). More precisely, it uses the 2D locations of mobile joints relative to their corresponding root joints. These inputs (dimension 2 × 12) are scaled to dimension 1024 by an affine layer, passed twice through the main processing unit (gray rectangle), and then scaled to output 3D joint-coordinates (dimension 3 × 12). **D** The main processing unit consists of two fully-connected layers of dimension 1024, followed by batch normalization, ReLU activation and dropout. These two layers are wrapped by a skip connection. **E** Absolute test error of LiftPose3D's one camera per side 3D pose prediction (gray) compared with the theoretical minimum of triangulation using two cameras per keypoint (white). These are quantified relative to triangulation using three cameras per keypoint. **F** Mean absolute error for cameras at −30° (blue), 0° (red), and 30° (yellow) with respect to the fly's mediolateral axis. **G** Average absolute test error of LiftPose3D's 3D pose predictions after the network is trained using data from flies performing optogenetically-induced backward walking (*MDN*, left), antennal grooming (*aDN*, middle), or spontaneously-generated behaviors (*PR*, right). **H** Two representative images from the OpenMonkeyStudio dataset. 2D poses are superimposed (black). **I** 3D poses obtained by triangulating up to 62 cameras (red lines) or using a single camera and LiftPose3D (dashed black lines). **J** Distribution of absolute errors for different body parts with respect to total body length. Violin plots represent Gaussian kernel density estimates with bandwidth 0.5, truncated at the 99th percentile and superimposed with the median (gray dot), 25th, and 50th percentiles (black box).

2.1 Reducing the number of cameras required for full 3D pose estimation

To date, 3D pose estimation in laboratory experiments has required the triangulation of multiple, synchronized camera views per keypoint [7, 16]. Using LiftPose3D, only one camera view per keypoint is needed. To illustrate this, we first applied LiftPose3D to a previously published tethered adult *Drosophila* dataset [7]. In this dataset, flies were recorded during optogenetically-induced behaviors like antennal grooming, and backward walking, as well as during spontaneously-generated behaviors including forward walking. Because in this system three cameras can view the same 15 keypoints on the left or right side of the animal (**Figure 1A**), the 3D coordinates of each keypoint can be triangulated using up to three 2D poses (**Figure 1B**, solid lines). We aimed to reduce the number of cameras needed for full (all limbs) 3D pose estimation from three to one camera per keypoint, while maintaining similar accuracy. To achieve this, we trained one LiftPose3D network against half-body poses visible from one camera on each side of the animal. We trained the network to predict the relative location of leg joints with respect to the thorax-coxa "root" joint on the same leg. For training, we sampled $\sim 2.5\times10^4$ frames from recordings of five flies and considered as inputs 2D poses from one camera on each side of the animal and as outputs their corresponding 3D ground truths obtained from three camera triangulation (**Figure 1C**; cameras 2 and 5). We tested the robustness of our LiftPose3D network predictions to variations in animal proportions using a test dataset consisting of $\sim 3.6\times10^3$ 2D-3D point pairs from two different animals. We computed the absolute error (AE) $e_j^{\text{te}} = ||(F(\mathbf{x}_c;\,\Theta^*))_j - \mathbf{Y}_{c,j}||_1$ for each joint j of the network's predictions as well as the average AE across all joints $e^{\text{te}} = (1/n)\sum_j e_j^{\text{te}}$ and observed that, using only one camera per keypoint (one on each side of the animal), LiftPose3D could predict full 3D poses with an accuracy comparable to triangulation using two cameras per keypoint (**Figure 1E** and **Video 1**). Although accuracy was high for all keypoints, the AE progressively increased for distal joints compared with proximal joints. For lifting, this is to be expected because the network predicts joint coordinates with respect to the thorax-coxa root joints; nearby, proximal joints move within a smaller kinematic volume. By contrast, triangulation obtains the 3D coordinates for each keypoint independently and, consequently, its error depends only on the accuracy of underlying 2D annotations. These are relatively uncorrelated with proximal-distal joint location.

We then tested the dependence of LiftPose3D's test error on camera angle by retraining the network for three different viewing arrangements (**Figure 1F**). These consisted of pairs of cameras arranged either along the mediolateral body axis (cameras 2 and 5) or ±30° from this axis (cameras 3 and 4, or cameras 1 and 6). Similarly excellent mean AE (~ 0.04 mm) were obtained for all three camera arrangements.

LiftPose3D prediction accuracy is expected to depend on the degree of overlap between training and test datasets. This is an especially critical factor in the low data regime characteristic of laboratory experiments. To probe this dependency, we trained LiftPose3D using pose data from only optogenetically-induced antennal grooming (aDN), or backward walking (MDN) while keeping the amount of training data (2.5×10^4 poses) fixed. As expected, the AE was higher for untrained optogenetically-induced and spontaneously-generated control behaviors (PR) than for test data including the same behaviors as in the training data (**Figure 1G**).

We further illustrate the value of these predicted 3D poses by comparing their underlying joint angles ($\alpha,\beta,\gamma,\omega$, **Figure S1**, red), with angles from 3D triangulated ground truth (**Figure S1**, blue), and angles from a 2D ventral projection in the x-y plane ($\alpha,\beta,\gamma,\omega$, **Figure S1**, green) during forward walking. Owing to the low mechanical compliance of the fly's exoskeleton (with the exception of the flexible tarsal segments) we could approximate the error of the predicted 3D joint angles by Monte Carlo sampling, assuming that the keypoints of the reconstructed 3D pose were drawn from a Gaussian distribution with variance estimated from the variation of leg segment lengths (see Materials and Methods). For all joint angles, we estimated low errors relative to the amplitude of variation during a swing-stance cycle, as our network learned and preserved body proportions—a remarkable fact given the absence of any skeletal constraints or without any temporal information.

We observed that 3D triangulated data (blue) were in close agreement with LiftPose3D's predicted poses (red). These data also illustrated several key advantages of studying joint angles derived from predicted 3D poses rather than from measured, but projected, 2D poses. First, in the front and hindlegs, the predicted coxa-femur 3D joint angles, β, were of larger amplitude than their projected 2D counterparts, β'. This would be expected since the action of these joints has a large out-of-plane component relative to the projected x-y plane during walking. Second, in the front leg, the predicted tibia-tarsus 3D joint angles, ω, were of smaller amplitude than their projected 2D counterparts, ω'. This could be explained by the fact that proximal joint rotations cause the movement of the whole leg, which can introduce spurious variations in the angles of distal joints when viewed from a projected plane. In other words, despite being nearly static, rotations upstream in the kinematic chain can introduce angular variations in the projected 2D plane. Taken together, these results illustrate how 2D projected joint angles can both obscure real dynamic variations and also introduce spurious correlations between limb degrees-of-freedom. By predicting 3D poses and deriving their underlying joint angles, LiftPose3D can help to decouple these physical degrees-of-freedom.

Because LiftPose3D maintained excellent accuracy irrespective of viewing angle (**Figure 1F**), we next asked how it would perform when predicting 3D pose in freely behaving animals viewed from many different camera angles. We addressed this question by training LiftPose3D to predict 3D poses of freely behaving macaque monkeys from the OpenMonkeyStudio dataset [8]. These data consist of 3D poses obtained by triangulating markerless 2D pose estimates [39] from 62 calibrated, synchronized, and distributed cameras (**Figure 1H**). After training the network with only 6'571 3D poses, we could lift 3D poses from test images—including macaques walking, climbing, and sitting—from any of the 62 cameras (**Figure 1I**), and with a relatively small body length-normalized AE (**Figure 1J**).

Taken together, these results demonstrate that LiftPose3D can reduce the number of cameras required to perform full and accurate 3D pose estimation with simple data preprocessing and a relatively small but diverse training dataset.

2.2 Predicting 3D pose in freely behaving animals with occluded keypoints

In freely behaving animals, keypoints are often missing from certain camera angles due to self-occlusions. Thus, for these images, only partial ground truth 3D annotations can be obtained by triangulation. We asked how the global nature of lifting—all keypoints are treated simultaneously—might be leveraged in these cases to reconstruct information lost by these occlusions and to predict full 3D poses.

To address this question, we built an experimental system consisting of a transparent enclosure physically coupled to a right-angle prism mirror. This allowed us to use a single camera placed beneath the enclosure to simultaneously record both the ventral and side views of a freely behaving fly at high spatial resolution ($\sim$114 px mm^{-1}) and at 100 Hz (**Figure 2A**). Similar systems have previously been used to record behavioral data from flies and mice [40, 41, 38]. We trained two DeeperCut networks in DeepLabCut [6] to obtain 2D joint coordinates from these ventral and side view images, respectively (**Figure 2A**). Having only two views meant that typically only those keypoints closer to the prism were simultaneously visible in both views and could be triangulated. With this partial 3D ground truth, it was *a priori* unclear if it would be possible to train a LiftPose3D network to predict full 3D poses.

Notably, in this prism-based system and unlike for tethered *Drosophila*, all limb keypoints are visible from the ventral view. This allowed us to simplify the prediction problem: we would attempt to use 2D poses from the ventral view to estimate the z-axis depth of occluded keypoints in the unseen side view. This is possible because the ventral and side views enclose right angles with respect to one another (i.e., are orthographic projections of the true 3D pose), and because long focal length cameras have negligible perspective effects. Another unique feature of the ventral view is that it allowed us to leverage the bilateral symmetry of the fly during network training. Specifically, by registering all images to create a dataset in which flies are only facing leftward (**Figure 2B**), estimating the z-axis depth became a regression problem of minimizing

$$\mathcal{J}_2(\Theta) = \sum_{j=1}^{n} \chi_{V_{\mathsf{side}}}(j)\, ||(F(x_j,\, y_j;\, \Theta))_j - z_j||_2^2\,, \tag{2}$$

where $\chi_{V_{\mathsf{side}}}(\cdot)$ is the indicator function of the set V_{side} of visible keypoints from the side camera (**Figure 2C**). Note that **Figure 2** presents all keypoints to the network simultaneously, but penalizes only those which are visible from the side view. This is a subtle but important distinction from **Figure 1**, and here allows the network to implicitly regress the unseen coordinates during training. With this data alignment, we found that LiftPose3D could predict the missing points during training and generate 3D joint predictions for every limb, including those occluded in the prism's side view

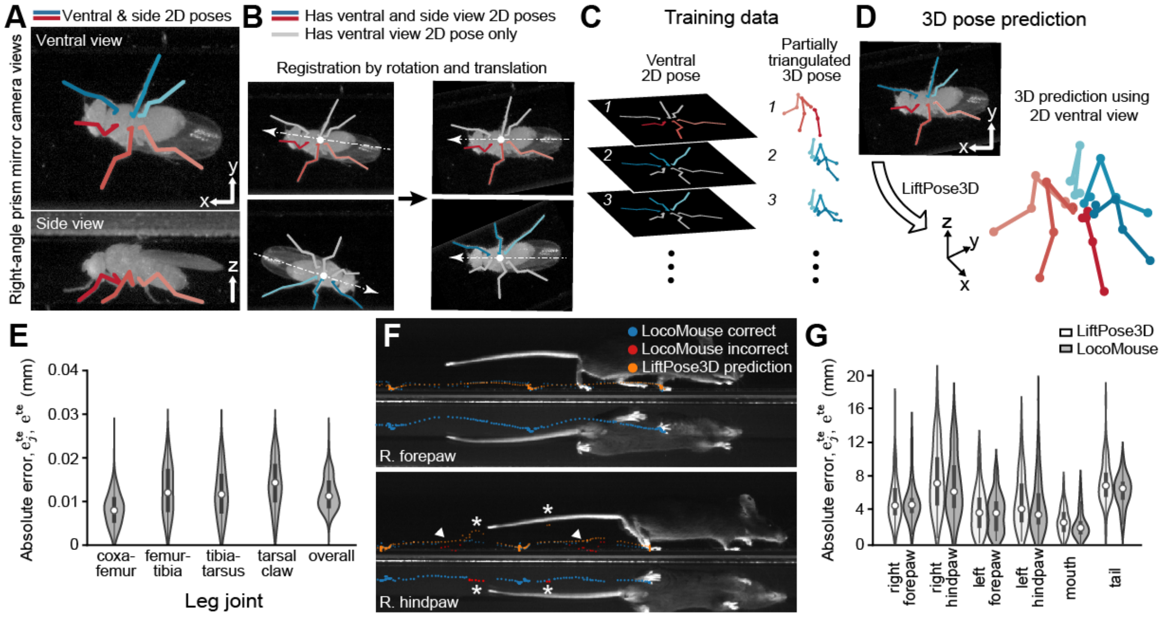

Figure 2: **LiftPose3D performs 3D pose estimation on freely behaving animals with occluded keypoints. A** *Drosophila* behaving freely within a narrow, transparent enclosure. Using one camera and a right-angle prism mirror, both ventral (top) and side (bottom) views are visible. 2D poses are tracked using two separately trained deep networks for each view (colored lines). **B** Keypoints near the prism mirror (red and blue) can be tracked in both views and triangulated. The remaining keypoints (gray) are only visible in the ventral view and thus have no 3D triangulated ground truth. To obtain triangulated ground truth examples for both sides of the bilaterally symmetric fly, we align the orientation and position of all animals. **C** Training data thus consists of a set of full ventral view 2D poses and their corresponding partially triangulated 3D poses. **D** Following training with these aligned 2D-3D ground truth data, LiftPose3D can be used to predict 3D poses for new ventral view 2D pose data. **E** Joint-wise and overall absolute errors of the network's 3D pose predictions for freely behaving *Drosophila*. **F** A similar data preprocessing approach can be used to lift ventral view 2D poses of mice walking within a narrow enclosure and tracked using the LocoMouse software. LocoMouse (blue and red) and LiftPose3D (orange) pose trajectories are shown for the right forepaw (top) and hindpaw (bottom) for one walking epoch. Arrowheads indicate where LiftPose3D lifting of the ventral view can be used to correct LocoMouse side view tracking errors (red). Asterisks indicate where LocoMouse ventral view tracking inaccuracies (red) disrupt LiftPose3D's side view predictions (yellow). **G** Absolute errors of LiftPose3D and LocoMouse side view predictions for six keypoints with respect to a manually-annotated ground truth.

(**Figure 2D** and **Video 2**). Furthermore, lifted 3D positions from the test data were in excellent agreement with available triangulation-derived 3D positions: the AE for test data (**Figure 2E**) were better than those obtained using four cameras to triangulate full 3D poses in tethered flies (**Figure 1E**). Thus, LiftPose3D can be used to estimate 3D poses in freely behaving animals with occluded keypoints that cannot be triangulated.

A further opportunity suggested by these results is that one might use lifting to identify and potentially correct inaccurate 3D poses obtained using other tracking approaches. We examined this possibility by lifting a previously published dataset consisting of freely behaving mice traversing a narrow corridor and recorded in two views, also using a mirror [38]. These images were previously annotated using the LocoMouse tracking software. Although many poses were correctly annotated by LocoMouse (**Figure 2F**, top, blue), others were either missing or incorrect (**Figure 2F**, bottom, red). We reasoned that because it learns consistent kinematic statistics, LiftPose3D might generate correct poses in place of these outliers. To test this, we preprocessed these mouse data as we did for the *Drosophila* prism mirror dataset, trained a LiftPose3D network, and predicted the relative 3D positions of six major keypoints tracked by LocoMouse—the four paws, the proximal tail, and the nose—from the ventral view and with respect to a virtual "root" keypoint placed on the ground midway between the nose and the tail.

During testing, LiftPose3D's ventral view-based predictions were in excellent agreement with the LocoMouse's side view tracking (**Figure 2E**). Both also exhibited cycloid-like kinematics between strides (**Figure 2F**). Remarkably, accurate 2D poses from our network could also be used to identify and correct side-view poses that were incorrectly labeled by LocoMouse (**Figure 2F**, bottom, white arrowheads). However, the accuracy of our network's predictions depended on the accuracy of ventral view 2D pose inputs. When these 2D poses were incorrectly labeled by LocoMouse, our network sometimes generated false side view predictions (**Figure 2F**, bottom, white asterisks). These cases were relatively infrequent and, overall, LiftPose3D performed as well as LocoMouse in predicting side view poses, when compared with manual human annotation (**Figure 2G**). These findings suggest that LiftPose3D can not only be used to generate full 3D poses in the face of self-occlusions, but can also be used to correct other tracking methods by cross-referencing lifted poses from multiple viewpoints.

2.3 Using domain adaptation to lift diverse experimental data when triangulation is impossible

At first glance it may appear that a laboratory must first generate a unique 2D-3D ground truth dataset to train a LiftPose3D network for their own experimental system. The possibility for domain adaptation—using training data from one system to predict 3D poses on test data for another system—is further hindered by the lack of openly available laboratory animal 2D-3D pose ground truth data compared with the enormous datasets available for human pose [42].

To address whether domain adaptation might be possible in the laboratory context, we applied our prism mirror 2D-3D training data to lift 2D data from two new target domains, both consisting of single ventral view camera recordings of freely behaving flies.

This single camera approach is the most widely used free behavior paradigm in the laboratory due to its simplicity, low-cost, and increased throughput: it has been applied to many organisms including *C. elegans* [43], larval zebrafish [44], larval *Drosophila* [45], adult *Drosophila* [46], and mice [47]. Although these recordings can be augmented with depth sensors [48, 49], such sensors cannot resolve very small laboratory animals, or reconstruct full 3D poses. Thus, 3D pose estimation of laboratory animals from a single 2D view remains an unsolved and highly desirable goal, with the potential to substantially enrich behavioral datasets and to improve downstream analysis.

To examine the efficacy of domain adaptation to this context, we first developed a new experimental system in which multiple flies could be filmed behaving freely within a square-shaped experimental arena with rounded corners. They were recorded at 80 Hz using a single camera with a view of their ventral body parts (**Figure 3A**). Importantly, in addition to being an entirely different experimental system from our prism mirror-based system, here the images were recorded at a much lower spatial resolution (26 px mm^{-1}). To accommodate the four-fold lower resolution of our target dataset, we perturbed the prism mirror-based 2D-3D pose training data using a Gaussian noise term with standard deviation equal to ~ 4 (**Figure 3B**)(see Materials and Methods). Because ventrally-viewed leg configurations during swing and stance phases are not clearly distinguishable from one another, to reconstruct realistic joint movements our network would have to first learn the postural relationships between each leg. Therefore, we had to ensure that joint identities were precisely matched across datasets. This was challenging because, in these lower resolution images, the coxa-femur joints are no longer visible. Thus, we could only label 24 visible keypoints using DeepLabCut. We then trained the LiftPose3D network with these augmented prism mirror-based data to predict the relative location of 18 leg joints with respect to their corresponding 6 thorax-coxa root joints. Prior to testing, we also aligned the target and training datasets to ensure statistical consistency (**Figure S3A**). This is an important step for domain adaptation across different camera systems and zoom factors.

Remarkably, we found that a network trained in this manner could predict physiologically realistic 3D poses in this new dataset using only ventral 2D poses (**Figure 3C** and **Video 4**). During walking, 2D tracking of the tarsal claws traced out stereotypical trajectories in the x-y plane (**Figure 3D**, top) [50] which corresponded to circular movements in the unmeasured x-z plane (**Figure 3D**, bottom). These predicted x-z cycles were of largest amplitude for the front, prothoracic legs, consistent with real kinematic measurements during forward walking [51].

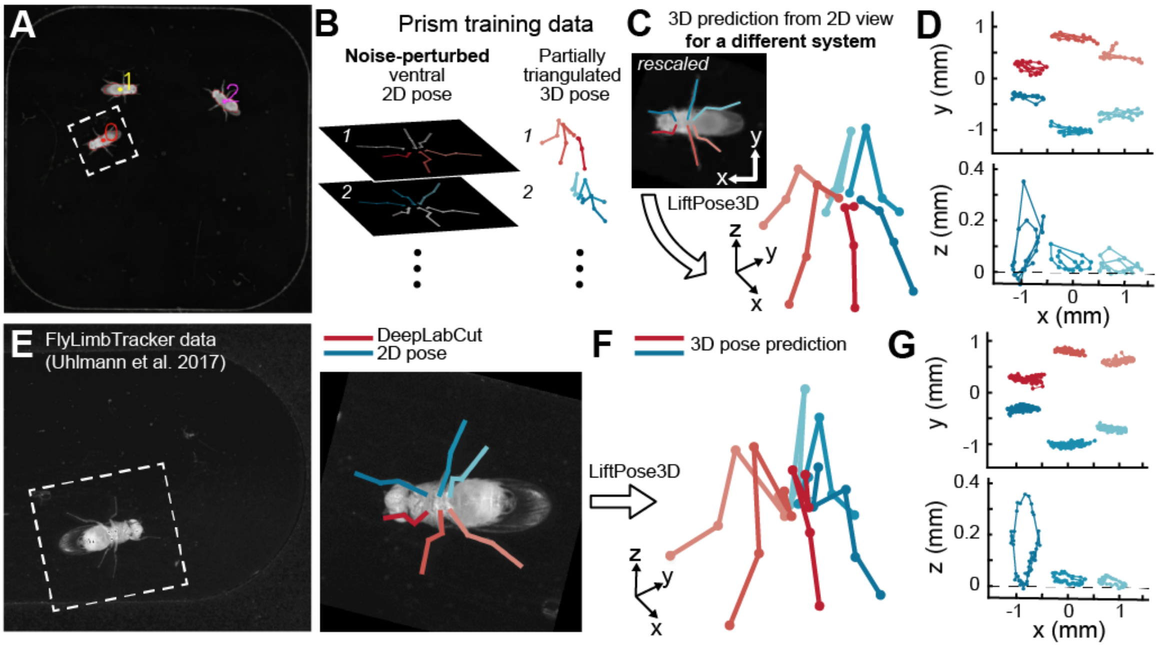

Figure 3: **A pretrained LiftPose3D network predicts 3D poses for diverse data and when triangulation is impossible. A** Freely behaving flies recorded from below using a single low-resolution camera. Following body tracking, a region-of-interest containing the fly is cropped and rotated to maintain a leftward orientation. 2D pose estimation is then performed for the 24 visible joints. **B** A LiftPose3D network is trained with *Drosophila* partial ground truth data from the prism mirror-based experimental system (Figure 2C). 2D poses from that prism training dataset are coarse-grained with additive noise to more closely match the lower resolution of 2D ventral camera images in this new experimental system. **C** Rescaled, low-resolution 2D ventral view poses can be input to this pretrained LiftPose3D network to output 3D poses. **D** These 3D poses permit the analysis of claw movements in the otherwise unobserved $x-z$ plane (bottom). **E** Published video data from [19] showing a freely behaving fly within a pill-shaped arena and recorded from below using one high-resolution camera. 2D pose estimation was performed for all 30 joints. Following tracking, a region-of-interest containing the fly was cropped and rotated to maintain a leftward orientation. The same LiftPose3D network trained in panel B—but without coarse-graining—was augmented to predict **F** 3D poses and **G** unobserved claw movements in the $x-z$ plane (bottom).

The ability to adapt training data from one domain to another also raises the exciting possibility that LiftPose3D might be used to 'resurrect' previously published 2D video data for new 3D kinematic analysis. To test this possibility, we applied our prism mirror-based training data to lift previously published high-resolution (203 px mm^{-1}) video data of a fly walking freely through a capsule-shaped arena [19](**Figure 3E**). Using a similar set of preprocessing steps as in the previous case (**Figure 3B,C**) including registration (**Figure S3B**) but not noise perturbation—the target data were of similarly high resolution as the training data—the LiftPose3D network could effectively predict 3D poses from this published dataset (**Figure 3F**). We again observed physiologically realistic cyclical movements of the pretarsi during forward walking (**Figure 3G**, bottom; **Video 5**). Thus, thanks to data augmentations permitting the adaptation of pretrained networks to new domains, LiftPose3D is an effective tool for performing 3D pose estimation on previously published 2D video data for which 3D triangulation would otherwise be impossible.

3 Discussion

Here we have introduced LiftPose3D, a deep neural network-based tool to dramatically simplify and enable 3D pose estimation in a wide variety of laboratory scenarios. Our approach uses the deep neural network of [33], originally designed for human-pose estimation, and introduces a series of innovations to input data preprocessing and training procedures. These extensions enable network training despite the corruption of ground truth 3D triangulation by self-occlusions, and reduces the 2D-3D training data required by several orders of magnitude. Our approach hinges upon registering (reorientating) training and test data: manipulations that allow the network to learn important symmetries in animal morphology and behavior. Further, we provide a modular software pipeline for data preprocessing, network training, 3D predictions, and visualization that is set-up via an intuitive configuration file which defines all species-specific features (e.g., number of keypoints and root joint identities). We illustrate how LiftPose3D reduces the number of cameras required for 3D pose estimation; from three to one on each side in the case of tethered flies, and from 62 to one in freely behaving macaques, while in both cases maintaining high accuracy comparable to triangulation. For freely behaving flies and mice observed from two views using a right-angle mirror, we demonstrate that LiftPose3D can estimate 3D poses despite self-occlusions and identify and correct keypoints that are mislabeled using other keypoint tracking approaches. Finally, we have demonstrated domain adaptation through a coarse-graining procedure used to train networks on one dataset that can then be generalized to other datasets when 3D ground truth is not available and spatial resolutions differ. This opens up the possibility of lifting 3D poses from a large corpus of previously published 2D video data for further kinematic analysis. The networks with the largest and most diverse training data, like that of the tethered fly—may already be sufficiently robust to accurately lift 2D to 3D pose in other laboratories. We make networks that have been trained on smaller 2D-3D pose ground truth datasets available for further training.

The LiftPose3D framework is general and can be applied with very few changes in data preprocessing to different laboratory animals, experimental systems, data acquisition rates, image resolutions, and 2D pose input sources including—as we demonstrate in this study—the stacked hourglass network of DeepFly3D [7] and DeepLabCut [6]. Nevertheless, several factors must be taken into consideration when optimizing LiftPose3D for new experimental systems. First, input 2D poses must be precise and accurate. Predicting depth from a 2D projection depends on comparing the projected lengths of body parts and, therefore, input poses must be sufficiently well-resolved to discriminate between 3D poses that have similar 2D projections. Second, prediction accuracy depends on the diversity of training data—i.e., measured behaviors. We caution that previously untrained behaviors may not be as accurately lifted using a pretrained network. In the future, we envision that robust lifting networks might be generated by a communal, inter-laboratory aggregation of a 2D-3D pose ground truth training datasets that include a variety of spontaneously generated and experimentally-induced behaviors. Third, although our aim was to develop a general tool with minimal experiment or animal-specific features, further work might improve LiftPose3D predictions for specific applications by bootstrapping to 3D body priors, thereby constraining the space of possible 3D poses [52, 53, 54, 55, 56]. Finally, lifting might also be improved by using a network that incorporates temporal information for data acquired at a constant frame rate [34].

We anticipate that LiftPose3D can already accelerate the successful adoption of 3D pose estimation in laboratory research by reducing the need for complex and expensive synchronized multi-camera systems, and arduous calibration procedures. This, in turn, will improve the fidelity and quality of behavioral kinematic data needed to understand how actions emerge from multi-scale biological processes ranging from gene expression to neural dynamics and biomechanics.

4 Materials and Methods

4.1 Obtaining 3D pose ground truth data by triangulation

To obtain the 3D ground truth coordinates $\mathbf{X}_j \in \mathbf{R}^3$ for joints $j = 1, \ldots, n$ from a set of 2D keypoints $\mathbf{x}_{c,j}$ in images acquired by the cameras $c = 1, \ldots, N$ we followed the procedure described in [7]. Let $\pi_c : \mathbb{R}^3 \mapsto \mathbb{R}^2$ be the projection function mapping from the 3D points in the global coordinate system to 2D points in a local coordinate system centered on camera c such that $\mathbf{x}_{c,j} = \pi_c(\mathbf{X}_j)$. The camera projection functions can be expressed as a composition $\pi_c = \text{proj}_{1,2} \circ \phi_c$ of a projection $\text{proj}_{1,2} : \mathbb{R}^3 \mapsto \mathbb{R}^2$ to the first two coordinates and an affine function $\phi_c : \mathbb{R}^3 \mapsto \mathbb{R}^3$ transforming global coordinates to camera-centered coordinates. The function π_c can be parameterized using the pinhole camera model [14]. Expressing $\mathbf{X}_j = (x_j^1, x_j^2, x_j^3)$ in homogeneous basis $\widehat{\mathbf{X}}_j = (x_j^1, x_j^2, x_j^3, 1)$, we have

$$\phi_c(\mathbf{X}_j) := \mathbf{C}_c \widehat{\mathbf{X}}_j^T \tag{3}$$

where $\mathbf{C}_c$ is the matrix corresponding to the camera transformation ϕ_c and can be written as

$$\mathbf{C}_c = \left(\begin{array}{c|c} \mathbf{R}_c & \mathbf{T}_c \\ \hline 0 & 1 \end{array}\right) . \tag{4}$$

where $\mathbf{I} \in \mathbb{R}^{3\times3}$ is the identity matrix, $\mathbf{R}_c \in \mathbb{R}^{3\times3}$ and $\mathbf{T}_c \in \mathbb{R}^3$ are the rotation and translation matrices, respectively.

Then, triangulation of the coordinate $\mathbf{X}_j$ of joint j with respect to π_c is equivalent to minimising the reprojection error, i.e., the discrepancy between the 2D camera coordinate, $\mathbf{x}_{c,j}$, and the 3D coordinate projected to the camera frame, $\pi_c(\mathbf{X}_j)$. Let V_c be the set of visible joints from camera c. Then the reprojection error of joint j reads

$$e_{\mathsf{RP}}\left(j; \{\pi_c\}\right) = \sum_c \chi_{V_c}(j) \left|\left|\mathbf{x}_{c,j} - \pi_c(\mathbf{X}_j)\right|\right|_2^2 , \tag{5}$$

where $\chi_{V_c}(\cdot)$ is the indicator function of set V_c of visible keypoints from camera c. Note that the camera projection functions π_c are *a priori* unknown and traditionally obtained via calibration using a patterned surface. To avoid manual calibration, we minimize the re-projection error for all joints while simultaneously optimising the camera parameters [7], a procedure known as bundle adjustment [14]

$$\min_{\pi_c,\, \mathbf{X}_j} \sum_j e_{\mathsf{RP}}\left(j; \{\pi_c\}\right) . \tag{6}$$

Given a set of 2D observations, we solve this using a second-order optimization method to obtain the 3D points and the camera matrix simultaneously. We refer the reader to [7] for further details.

4.2 LiftPose3D network architecture and optimization

The core LiftPose3D network architecture follows the network in [33] and is illustrated in **Figure 1C-D**. It is modified only for lifting the OpenMonkeyStudio dataset where we observed that removing dropout layers improved performance. The main module of the network includes two linear layers of 1024 nodes applying batch normalization [57], rectified linear units (ReLU, [58]), and dropout with 0.5 probability [59] as well as residual connections [60]. The inputs and outputs of each block are connected during each forward pass using a skip connection. The model contains 4×10^6 trainable parameters, which are optimized by stochastic gradient descent using the Adam optimizer [61]. In all cases, the parameters were set using Kaiming initialization [60] and the optimiser was run until convergence (typically between 100-500 epochs) with the following training hyperparameters: batch-size of 64, and an initial learning rate of 10^{-3} which was dropped by 4% every 10^5 iterations. We implemented optimisation in PyTorch on a desktop workstation running on an Intel Core i9-7900X CPU with 32 GB of DDR4 RAM, and a GeForce RTX 2080 Ti Dual O11G GPU.

4.3 Experimental systems and conditions

All adult *Drosophila melanogaster* experiments were performed on female flies raised at 25°C on a 12 h light/dark cycle at 2-3 days post-eclosion (dpe). DeepFly3D tethered fly data were taken from [7].

OpenMonkeyStudio macaque data were taken from [8]. LocoMouse mouse data were taken from [38]. FlyLimbTracker freely behaving fly data were taken from [19]. See these publications for detailed experimental procedures.

4.3.1 Freely behaving *Drosophila* recorded from two high-resolution views using one camera and a right-angle prism mirror

We constructed a transparent arena coupled to a right-angle prism mirror [40, 41]. The enclosed arena consists of three vertically stacked layers of 1/16" thick acrylic sheets laser-cut to be 15 mm long, 3 mm wide, and 1.6 mm high. Before each experiment, the arena ceiling and walls were coated with Sigmacote (Sigman-Aldrich, Merck, Darmstadt, Germany) to discourage animals from climbing onto the walls and ceilings. One side of the enclosure was physically coupled to a right-angled prism (Thorlabs PS915). The arena and prism were placed on a kinematic mounting platform (Thorlabs KM100B/M), permitting their 3D adjustment with respect to a camera (Basler acA1920-150um) outfitted with a zoom lens (Computar MLM3X-MP, Cary, NC USA). The camera was oriented vertically upwards below the arena to provide two views of the fly: a direct ventral view, and an indirect, prism mirror-reflected side view. The arena was illuminated by four Infrared LEDs (Thorlabs, fibre-coupled LED M850F2 with driver LEDD1B T-Cube and collimator F810SMA-780): two from above and two from below.

Before each experiment, wild-type (*PR*) animals were anaesthetized using CO_2 and left to acclimate in the enclosure for 10 min. Flies were then allowed to behave freely, without optogenetic stimulation. To elicit locomotor activity, the platform was acoustically and mechanically stimulated using a mobile phone speaker. Recordings were made of 4 flies for 40 min each at a frame rate of 100 Hz, capturing 700 × 1792 px images (equivalent to 112 px mm^{-1}).

4.3.2 Freely behaving *Drosophila* recorded from one ventral view at low-resolution

We constructed a square arena consisting of three vertically stacked layers of 1/16" thick acrylic sheets laser-cut to be 30 mm long, 30 mm wide, and 1.6 mm high. This arena can house multiple flies at once, increasing throughput at the expense of spatial resolution (26 px mm^{-1}). Before each experiment the arena ceiling was coated with 10 uL Sigmacote (Sigman Aldrich, Merck, Darmstadt Germany) to discourage animals from climbing onto the ceiling. A camera (pco.panda 4.2 M-USB-PCO, Gloor Instruments, Switzerland, with a Milvus 2/100M ZF.2 lens, Zeiss, Switzerland) was oriented with respect to a 45 degree mirror below the arena to capture a ventral view of the fly. An 850 nm infrared LED ring light (CCS Inc. LDR2-74IR2-850-LA) was placed above the arena to provide illumination.

To record optogenetically-elicited backward walking, we used MDN>CsChrimson flies (*20x UAS-CsChrimson; VT50660-AD-GAL4; VT44845-DBD-GAL4*). MDN>CsChrimson animals were raised on food with all-trans-retinal for 24 hrs. Prior to experiments, they were anaesthetized in ice-cooled vials for 5 min. Then, three flies were placed into the arena and allowed to acclimate for 15-30 minutes. Periods of optogenetic stimulation were interspersed with periods of spontaneous behaviour. Here we focused only on spontaneously generated forward walking.

Flies were recorded at 80 Hz to generate 832 × 832 px images (equivalent to 26 px mm^{-1}). The positions and orientations of individual flies were tracked using custom software including a modified version of Tracktor [62]. Using these data, a 138 × 138 px image was cropped around each fly and rotated to reorient flies leftward for subsequent analyses.

4.4 2D pose estimation

DeepFly3D 2D poses were taken from [7]. OpenMonkeyStudio 2D poses were taken from [8]. LocoMouse 2D poses were taken from [38]. See these publications for detailed 2D pose estimation procedures.

4.4.1 2D pose estimation of freely behaving flies recorded in two views using a right-angle prism mirror

Data acquired from a single camera were first split into ventral and side view images. We hand-annotated the location of all 30 leg joints (five joints per leg) on 640 images with a ventral view and up to 15 visible unilateral joints on 640 images of the side view. We used these manual annotations to train two separate DeepLabCut [6] 2D pose estimation networks (RMS errors for training and testing were 0.02 mm and 0.04 mm for ventral and side views, respectively). Whereas ventral view images could be used to predict 2D pose for all 30 leg joints, from the side view at most 15 joints were visible when the fly was parallel to the prism. Typically fewer keypoints were visible due to rotations of the fly within the enclosure. We removed images in which DeepLabCut incorrectly annotated keypoints as well as images in which flies were climbing the enclosure walls (thus exhibiting large yaw and roll orientation angles). To exclude these images, we ignored those with a confidence threshold below 0.95, and those for which the x-coordinate between the lateral and ventral views differed by more than 10 px.

4.4.2 2D pose estimation of freely behaving flies recorded in one ventral view using a single camera

FlyLimbTracker [19] data were not sufficiently well-annotated. Thus, we replaced these annotations by manually annotating all 30 keypoints using DeepLabCut's graphical user interface.

For newly acquired low-resolution ventral view single camera data, we trained a DeepLabCut [6] 2D pose estimation network. Due to the low resolution of images, the coxa-femur joints were not distinguishable, therefore, we treated the thorax-coxa and coxa-femur joints as a single entity. We manually annotated 160 images with the location of four landmarks per leg: the thorax-coxa-femur entity, the femur-tibia joint, the tibia-tarsus joint, and the claw. We then trained a DeepLabCut network to predict the 2D coordinates of the 24 landmarks in the legs from the ventral view.

4.5 Training the LiftPose3D network

An important step in constructing LiftPose3D training data is to choose r root joints, which serve as the origins of a coordinate system to which the distance of a subset of keypoints are considered (see the specific use cases below for how these root joints were selected).

The training dataset consisted of input-output pose pairs $(\mathbf{x}_c^{\mathsf{tr}}, \mathbf{X}^{\mathsf{tr}})$ with dimensionality equal to the number of keypoints visible from a given camera c minus the number of root joints r, namely $\mathbf{x}_c^{\mathsf{tr}} \in \mathbb{R}^{2(|V_c|-r)}$ and $\mathbf{X}^{\mathsf{tr}} \in \mathbb{R}^{3(|V_c|-r)}$. Then, the training data was standardized with respect to the mean and standard deviation, μ_j, σ_j, of a given keypoint across all poses. Any slight differences in implementation were due to the preprocessing of 2D pose training data and are mentioned below.

4.5.1 Tethered *Drosophila melanogaster*

Of 38 original keypoints in [7], here we focused on the 30 leg joints because the depth of antennal and abdominal keypoints had very low variability. Specifically, for each leg we estimated 3D position for the thorax-coxa, coxa-femur, femur-tibia, and tibia-tarsus joints and the tarsal tips (claws). Thus, the training data consisted of input-output coordinate pairs $(\mathbf{x}_c^{\mathsf{tr}} + \epsilon, \mathbf{X}^{\mathsf{tr}})$ for $n = 24$ (30 minus six thorax-coxa root joints) joints and pairs of cameras $c = \{(3,4), (2,5), (1,6)\}$. Here $\mathbf{x}_c^{\mathsf{tr}} \in \mathbb{R}^{48}$ are 2D input joint keypoints acquired from camera c and $\mathbf{X}^{\mathsf{tr}} \in \mathbb{R}^{72}$ are 3D ground truth coordinates obtained from DeepFly3D by triangulating 2D coordinates from all six cameras. Furthermore, $\epsilon \in \mathbb{R}^{48}$ is an additive noise term, each with zero-mean Gaussian components having standard deviations equal to the fluctuation in the 2D stacked hourglass prediction over poses when the fly is stationary. We found that the additive noise term stabilizes the network's convergence during training (**Figure S2A**) and reduces uncertainty in lifted 3D joint positions. To maintain consistency for calculations of absolute error, triangulation was performed using the same set of 2D poses that were used to train the LiftPose3D network.

4.5.2 Freely behaving macaque monkeys

The OpenMonkeyStudio dataset [8] consists of images of freely behaving monkeys inside a 2.45×2.45×2.75 m arena in which 62 cameras are uniformly distributed horizontally at two heights along the arena perimeter. We extracted five experiments (7, 9, 9a, 9b and 11) consisting of 7'280 nonconsecutive frames. The 3D poses for these frames were composed of 13 keypoints (nose, neck, tail, hips, shoulders, hands, knees and feet). We removed 82 frames with unrealistic bone lengths (hundreds of meters long) or large reprojection errors (more than 100 pixels) resulting in a total of 7'198 poses. Since 2D pose annotations were not available for all cameras, we augmented this dataset by projecting triangulated 3D poses onto cameras lacking 2D annotation using the provided camera matrix.

We trained LiftPose3D to predict all 13 joints from 2D poses acquired from any of the 62 cameras. For training we used experiments 7, 9a, 9b, 11 consisting of 6'571 3D poses and their corresponding 2D poses from 62 cameras. The poses of experiment 9 were used as a test dataset. We restricted our analysis to 337 images in the test set, in which macaques were walking on the ground.

Before training, we preprocessed the data using the following steps. First, to reduce the complexity of the data, we removed the fish-eye lens related distortions of 2D poses using the radial distortion parameters obtained from camera calibration. Second, we normalized each 2D pose to unit length, by dividing it by its Euclidean norm. Third, to reduce the large scale variability of the OpenMonkeyStudio annotations (animals ranged between 5.5 and 12 kg), following the convention in human 3D pose estimation, we normalized the 3D pose scale with respect to bone lengths [42]. This was required since regressing a human/object scale from 2D pose has previously been shown to be challenging, even with a large annotated dataset [63]. Fourth, following the OpenMonkeyStudio convention, we set the neck joint as the root joint during training. We compare our absolute errors to the total body length, calculated as the sum of the mean lengths of the nose-neck, neck-hip, hip-knee, knee-foot joints pairs. Over multiple iterations, we observed rapid convergence of our trained network (**Figure S2B**).

4.5.3 Freely behaving mice and adult *Drosophila* recorded from two views using a right-angle mirror

Freely behaving mouse data [38] consisted of recordings of animals traversing a 66.5 cm long, 4.5 cm wide, and 20 cm high glass corridor. A 45° mirror was used to obtain both ventral and side views with a single camera beneath the corridor. Movies were collected at 400 Hz and at a spatial resolution of 1440 × 250 pixels (equivalent to 2.5 px mm^{-1}). 2D keypoint positions were previously tracked using the LocoMouse software [38]. These data consisted of 39'680 images with incomplete 3D poses triangulated from the ventral view, where all keypoints were visible, and the side view, where some keypoints were occasionally invisible.

For both the *Drosophila* and mouse datasets, side view keypoints distal to the camera were intermittently occluded by the animal's body. Thus, taking a simplistic approach, after training with this unilateral ground truth data, lifting from the ventral view would only be able to recover keypoints on the proximal half of the animal. Here we significantly modified data preprocessing to enable lifting across both the proximal and the occluded, distal side of the animal. Specifically, we realigned/rotated all animals to face leftward along the horizontal axis in the ventral view, thereby generating ground truth data for all leg joints across the entire dataset. Thus, although there is still only partial 3D pose ground truth for each image (for the proximal, fully visible half of the animal) we forced the lifting function f to predict the entire pose. This is made possible because the realignment step masks from the network which data, among all of the input to f, is visible and contains 3D ground truth annotations.

In other words, by combining the proposed alignment and partial 3D pose supervision, the training dataset includes coordinate pairs $(\mathbf{x}^{\mathsf{tr}}_{\mathsf{ventral}} + \epsilon, \mathbf{z}^{\mathsf{tr}}_{\mathsf{side}})$, with ϵ as before, $\mathbf{x}^{\mathsf{tr}}_{\mathsf{ventral}} = \{(x_j, y_j) : j \in V_{\mathsf{side}}\} \in \mathbb{R}^{2|V_{\mathsf{side}}|}$ are the coordinates of DeepLabCut annotated 2D keypoints from the ventral viewpoint and $\mathbf{z}^{\mathsf{tr}}_{\mathsf{side}} = \{z_j : j \in V_{\mathsf{side}}\} \in \mathbb{R}^{|V_{\mathsf{side}}|}$ are the corresponding z-axis depth coordinates, for joints visible from the side view for a given frame. For *Drosophila* data, the training and test datasets consisted of 11'840 (acquired from three animals) and 3'456 frames (acquired from one distinct animal), respectively. The mouse training and test datasets consisted of 198'336 and 39'680 and frames, respectively, acquired from 23 male and 11 female mice. Both networks converged in fewer than 300 training epochs (**Figure S2C,D**).

4.5.4 Freely behaving adult *Drosophila melanogaster* recorded from one ventral camera view

For both the newly acquired low-resolution and previously published high-resolution [19] images of freely behaving flies taken using one ventral view camera, we trained a LiftPose3D network on partial ground truth data acquired from the prism mirror system. For the high-resolution data, we considered the thorax-coxa joints as roots. For the low resolution data coxa-femur joints were imperceptible, hence the thorax-coxa joints were selected as roots. As before, we focused on predicting the relative location of the remaining mobile joints (24 and 18 keypoints, respectively) with respect to their associated root joints. The training dataset consisted of coordinate pairs ($\mathbf{x}^{\mathsf{tr}}_{\mathsf{ventral}} + \epsilon + \eta$, $\mathbf{z}^{\mathsf{tr}}_{\mathsf{side}}$) where $\mathbf{x}^{\mathsf{tr}}_{\mathsf{ventral}}$, ϵ, $\mathbf{z}^{\mathsf{tr}}_{\mathsf{side}}$ were chosen to represent the annotated ventral coordinates, joint-dependent noise and z-axis depth for the visible joints, as before. Meanwhile, η was a novel noise term, which we describe below.

The training and test data were augmented to accomplish domain adaptation: lifting new data with the prism system training data. First, for the low-resolution dataset, a zero-mean Gaussian noise term with a joint-independent standard deviation of 4 px, η, was added during training. The role of this noise term was to account for the keypoint position degeneracy inherent in the transformation from high-resolution prism training data to lower-resolution testing data. This term effectively coarse-grained the network's spatial resolution, accounting for the 4-fold lower resolution of the low-resolution single camera ventral view system compared with the right-angle prism mirror system. For the high resolution dataset this noise term was set to zero. Using these modifications the networks displayed stable convergence within approximately 300 training epochs (**Figure S2E, F**).

Second, following training, we preprocessed the test data 2D poses derived from both the low- and high-resolution images by matching their data distributions to that of the prism-mirror dataset. To achieve this, we performed procrustes analysis to find the optimal affine transformation (rotation, translation and scaling) that maps the average root joint positions across poses in the test dataset to those in the prism-mirror dataset. After this scaling, the distributions of 2D joint coordinates matched across datasets (**Figure S3A, B**).

4.6 Deriving joint angles and performing error estimates

Consider three consecutive joints in the kinematic chain of one leg with coordinates $\mathbf{u}$, $\mathbf{v}$, $\mathbf{w}$, which can live in 2D space when obtained by 2D pose estimation or in 3D space when obtained by triangulation, or by LiftPose3D reconstruction. Then, vectors $\mathbf{s}_1 = \mathbf{u} - \mathbf{v}$ and $\mathbf{s}_2 = \mathbf{u} - \mathbf{w}$ describe adjacent bones and their enclosed angle is found by the cosine rule, $\cos^{-1}(\,\mathbf{s}_1 \cdot \mathbf{s}_2/(||\mathbf{s}_1||\,||\mathbf{s}_2||)\,)$.

With the exception of the tarsus, the fly's exoskeleton moves in a rigid manner. This permits the estimation of errors in the lifted joint angles based on fluctuations of predicted bone lengths. To do so, we assume that $\mathbf{u}$, $\mathbf{v}$, $\mathbf{w}$ are drawn from independent Gaussian distributions centered around the estimated coordinate with standard deviation equal to the variation of the bone lengths $||\mathbf{s}_1||$ and $||\mathbf{s}_2||$. The distribution of joint angles for any given pose was estimated by Monte Carlo sampling using 5×10^3 samples.

4.7 Code and data availability

Code can be found at:
https://github.com/NeLy-EPFL/LiftPose3D
Data are available upon request.

5 Supplementary Figures

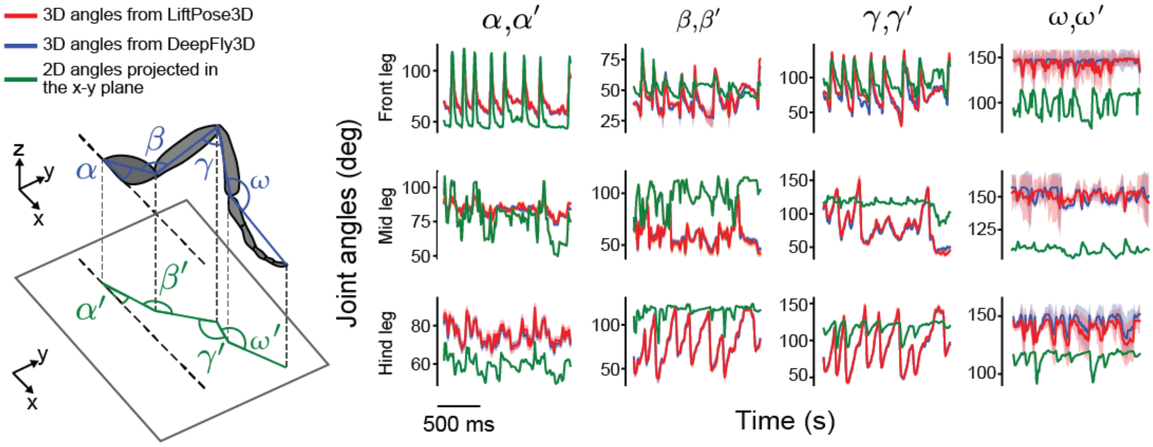

Figure S1: **Joint angles resulting from lifting compared with 3D triangulated ground truth and 2D projections.** Joint angles α, β, γ, and ω for the front, mid, and hind left legs during forward walking. Shown are angles computed from 3D triangulation using DeepFly3D (blue), LiftPose3D predictions (red), and ventral 2D projections α', β', γ, and ω (green). The mean (solid lines) and standard deviation of joint error distributions (transparency) are shown. Joint angles were computed by Monte Carlo sampling and errors were computed by taking the fluctuation in bone lengths.

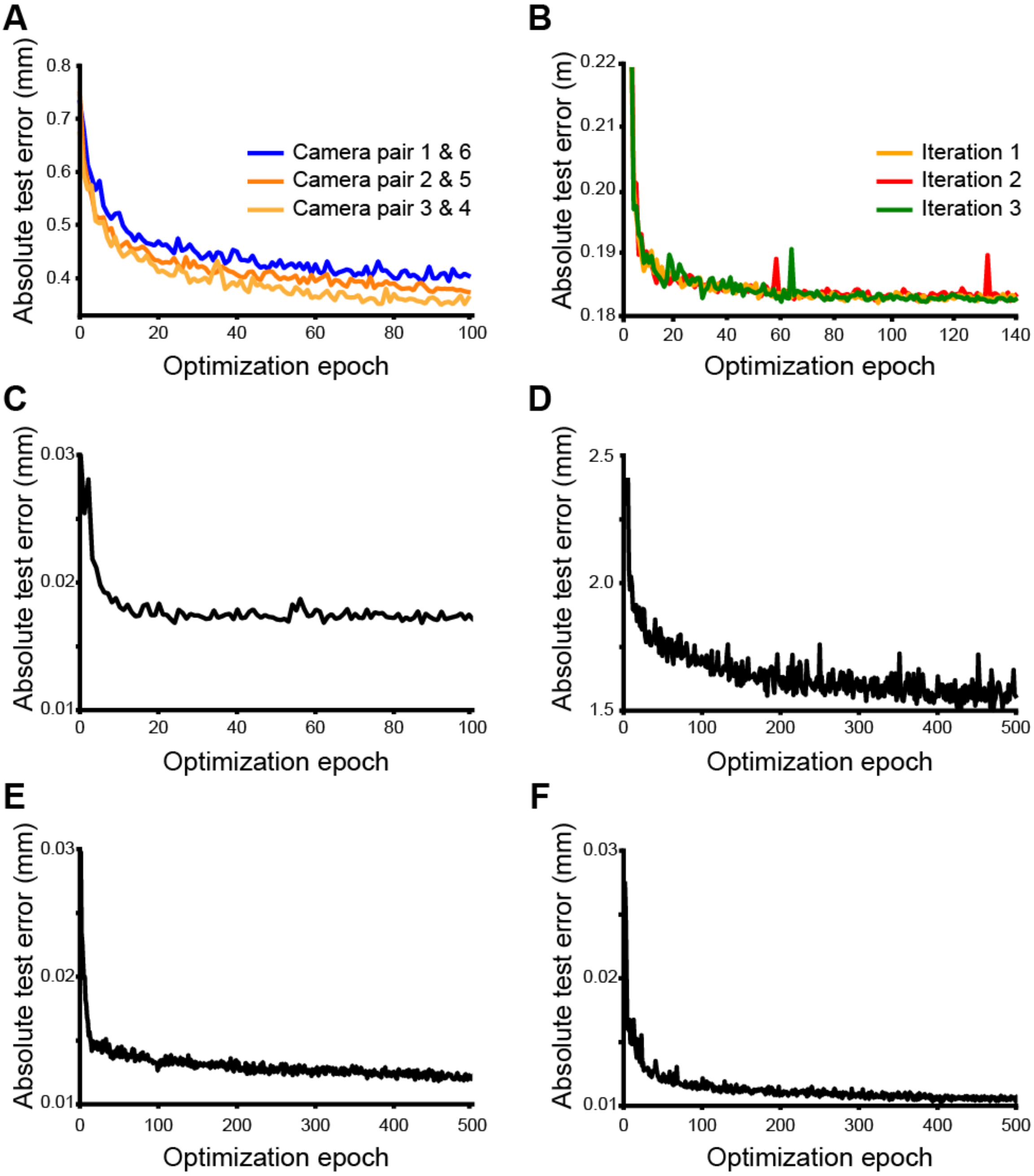

Figure S2: **Test error convergence of the LiftPose3D network applied to a variety of datasets.** Shown are the absolute test errors of LiftPose3D for all joints as a function of optimization epoch for **A** two-camera data of *Drosophila* on a spherical treadmill (each color denotes a different pair of diametrically opposed cameras), **B** the OpenMonkeyStudio dataset (each color denotes a different training run), **C** single-camera data of *Drosophila* behaving freely in the right-angle prism mirror system, **D** the LocoMouse dataset, **E** low-resolution, one-camera data of *Drosophila* behaving freely in a rounded square arena, and **F** the FlyLimbTracker dataset.

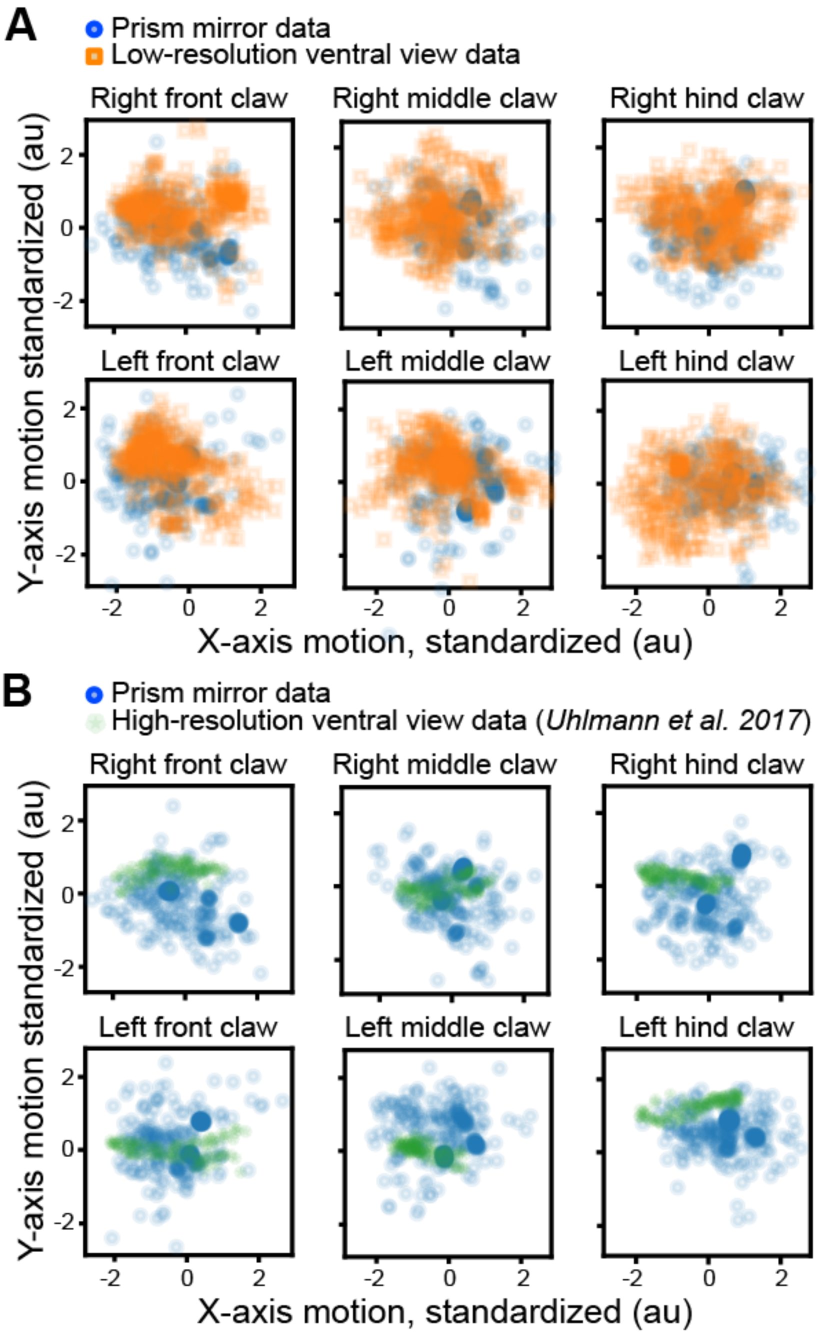

Figure S3: **Distribution of the claw positions across datasets following inter-domain alignment.** Point clouds show the distribution of six tarsal claw coordinates following affine transformation to match the mean position across datasets. Data have been standardized by the mean and standard deviation of the point clouds. Shown are claw positions for the prism mirror training data (blue) and the **A** low-resolution ventral view images (orange), or the **B** published FlyLimbTracker high-resolution ventral-view images (green).

6 Supplementary Videos

Video 1: 3D pose estimation of tethered animals on a spherical treadmill using only two side-view cameras. Videos obtained from cameras 2 **(top-left)** and 5 **(bottom-left)**. DeepFly3D-derived 2D poses are superimposed. Orange circle indicates that the optogenetic stimulation LED light is on, activating MDNs to elicit backward walking. **(right)** 3D poses obtained by triangulating six camera views using DeepFly3D (solid lines), or lifting two camera views using LiftPose3D (dashed lines).
https://www.dropbox.com/s/23xzzfu0tkg6frs/video_1.mp4?dl=0

Video 2: 3D pose estimation of freely behaving *Drosophila* when triangulation is only partially possible. Single camera images of the ventral **(top-left)** and side **(bottom-left)** views. DeepLabCut-derived 2D poses are superimposed. **(right)** 3D poses obtained by triangulating partially available multi-view 2D poses (solid lines), or by lifting the ventral 2D pose using LiftPose3D (dashed lines).
https://www.dropbox.com/s/wo62cx4gsttyso2/video_2.mp4?dl=0

Video 3: 3D pose estimation of freely behaving mice when triangulation is only partially possible. Side **(top-left)** and ventral **(bottom-left)** views of a freely walking mouse. Superimposed are keypoints on the paws, mouth, and proximal tail tracked using the LocoMouse software (blue circles). Using only the ventral view 2D pose, a trained LiftPose3D network can accurately track keypoints in the side view (orange circles).
https://www.dropbox.com/s/1irnms0txoci9as/video_3.mp4?dl=0

Video 4: 3D pose estimation for low-resolution videos of freely behaving flies when triangulation is impossible. (top) Three freely behaving *Drosophila* in a rounded square arena and recorded ventrally using a single low-resolution camera. Of these, fly 0 is tracked, cropped, and rotated leftward. Superimposed are 2D poses for 24 visible joints. (bottom) 3D poses lifted from ventral view 2D poses ($x - y$ plane) permit analysis of leg kinematics in the otherwise unobserved $x - z$ plane.
https://www.dropbox.com/s/00iy1r02urd6m0n/video_4.mp4?dl=0

Video 5: 3D pose estimation of previously published ventral view videos of freely behaving flies when triangulation is impossible. (top) Video from [19] of a freely behaving fly within a pill-shaped arena and recorded ventrally using a single high-resolution camera. **(bottom-left)** Following tracking, a region-of-interest containing the fly was cropped and rotated to maintain a leftward orientation. Superimposed are 2D poses estimated for 24 visible joints. **(bottom-middle)** 3D poses obtained by lifting ventral view 2D poses. **(bottom-right)** 3D poses lifted from ventral view 2D poses (top) permit analysis of leg kinematics in the otherwise unobserved $x - z$ plane (bottom).
https://www.dropbox.com/s/cfgjv0ugzdkwnu1/video_5.mp4?dl=0

7 Funding

PR acknowledges support from an SNSF Project grant (175667), an SNSF Eccellenza grant (181239), and an EPFL iPhD grant. AG acknowledges support from an HFSP Cross-disciplinary Postdoctoral Fellowship (LT000669/2020-C). SG acknowledges support from an EPFL SV iPhD Grant. DM holds a Marie Curie EuroTech postdoctoral fellowship and acknowledges that this project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 754462. VLR acknowledges support from the Mexican National Council for Science and Technology, CONACYT, under the grant number 709993. PF acknowledges support from an EPFL iPhD grant.

8 Acknowledgments

We thank Megan Carey (Champalimaud Centre for the Unknown, Lisbon, Portugal) for the LocoMouse dataset.

9 Author Contributions

A.G. - Conceptualization, methodology, software (LiftPose3D), hardware (*Drosophila* prism mirror system), formal analysis (all *Drosophila*, and LocoMouse datasets), investigation (prism mirror *Drosophila* experiments), data curation, writing—original draft, writing—review & editing, visualization.
S.G. - Conceptualization, methodology, software (LiftPose3D), formal analysis (DeepFly3D joint angle analysis. LocoMouse and OpenMonkeyStudio datasets), data curation, writing—original draft, writing—review & editing, visualization.
M.A. - Methodology, software (LiftPose3D), formal analysis, data curation, writing—review & editing.
D.M. - Investigation (low-resolution *Drosophila* experiments), writing—review & editing.
V.L.R. - Software and hardware (low-resolution *Drosophila* ventral view system), data curation, writing—review & editing.
H.R. - Conceptualization, writing—review & editing.
P.F. - Writing—review & editing, funding acquisition.
P.R. - Conceptualization, hardware (*Drosophila* prism mirror system), resources, writing—original draft, writing—review & editing, supervision, project administration, funding acquisition.

10 Competing interests

The authors declare that no competing interests exist.

References

[1] Dombeck, D. A., Khabbaz, A. N., Collman, F., Adelman, T. L. & Tank, D. W. Imaging large-scale neural activity with cellular resolution in awake, mobile mice. *Neuron* **56**, 43 – 57 (2007).

[2] Seelig, J. D. *et al.* Two-photon calcium imaging from head-fixed *Drosophila* during optomotor walking behavior. *Nature Methods* **7**, 535–540 (2010).

[3] Churchland, M. M. *et al.* Neural population dynamics during reaching. *Nature* **487**, 51–56 (2012).

[4] Chen C. L., Hermans L. *et al.* Imaging neural activity in the ventral nerve cord of behaving adult *Drosophila*. *Nature Communications* **9**, 4390 (2018).

[5] Pereira, T. D. *et al.* Fast animal pose estimation using deep neural networks. *Nature Methods* **16**, 117–125 (2019).

[6] Mathis, A. *et al.* DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience* **21**, 1281–1289 (2018).

[7] Günel, S. *et al.* DeepFly3D, a deep learning-based approach for 3D limb and appendage tracking in tethered, adult *Drosophila*. *eLife* **8**, 3686 (2019).

[8] Bala, P. C. *et al.* OpenMonkeyStudio: Automated markerless pose estimation in freely moving macaques. *bioRxiv* (2020).

[9] Newell, A., Yang, K. & Deng, J. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision (ECCV)* (2016).

[10] Graving, J. M. *et al.* DeepPoseKit, a software toolkit for fast and robust animal pose estimation using deep learning. *eLife* **8**, e47993 (2019).

[11] Fang, H.-S., Xie, S., Tai, Y.-W. & Lu, C. RMPE: Regional multi-person pose estimation. In *IEEE International Conferene on Computer Vision (ICCV)* (2017).

[12] Wei, S.-E., Ramakrishna, V., Kanade, T. & Sheikh, Y. Convolutional pose machines. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[13] Cao, Z., Simon, T., Wei, S.-E. & Sheikh, Y. Realtime multi-person 2d pose estimation using part affinity fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017).

[14] Hartley, R. & Zisserman, A. *Multiple View Geometry in Computer Vision* (Cambridge University Press, USA, 2003), 2 edn.

[15] Karashchuk, P. *et al.* Anipose: a toolkit for robust markerless 3d pose estimation. *bioRxiv* (2020).

[16] Nath, T., Mathis, A., Chen, A. C., Bethge, M. & Mathis, M. W. Using DeepLabCut for 3d markerless pose estimation across species and behaviors. *Nature Protocols* **14**, 2152–2176 (2019).

[17] Gaudry, Q., Hong, E. J., Kain, J., de Bivort, B. L. & Wilson, R. I. Asymmetric neurotransmitter release enables rapid odour lateralization in *Drosophila*. *Nature* **493**, 424–428 (2013).

[18] Isakov, A. *et al.* Recovery of locomotion after injury in *Drosophila melanogaster* depends on proprioception. *Journal of Experimental Biology* **219**, 1760–1771 (2016).

[19] Uhlmann, V., Ramdya, P., Delgado-Gonzalo, R., Benton, R. & Unser, M. FlyLimbTracker: An active contour based approach for leg segment tracking in unmarked, freely behaving *Drosophila*. *PLoS One* **12**, e0173433 (2017).

[20] DeAngelis, B. D., Zavatone-Veth, J. A. & Clark, D. A. The manifold structure of limb coordination in walking *Drosophila*. *eLife* **8**, 137 (2019).

[21] Lee, H.-J. & Chen, Z. Determination of 3d human body postures from a single view. *Computer Vision, Graphics, and Image Processing* **30**, 148 – 168 (1985).

[22] Taylor, C. J. Reconstruction of articulated objects from point correspondences in a single uncalibrated image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2000).

[23] Chen, C. & Ramanan, D. 3d human pose estimation = 2d pose estimation + matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017).

[24] Gupta, A., Martinez, J., Little, J. J. & Woodham, R. J. 3d pose from motion for cross-view action recognition via non-linear circulant temporal encoding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2014).

[25] Sun, J. J. *et al.* View-invariant probabilistic embedding for human pose. In *arXiv* (2019).

[26] Nibali, A., He, Z., Morgan, S. & Prendergast, L. 3d human pose estimation with 2d marginal heatmaps. In *IEEE Winter Conference on Applications of Computer Vision (WACV)* (2019).

[27] Zhao, L., Peng, X., Tian, Y., Kapadia, M. & Metaxas, D. N. Semantic graph convolutional networks for 3d human pose regression. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

[28] Iskakov, K., Burkov, E., Lempitsky, V. & Malkov, Y. Learnable triangulation of human pose. In *International Conference on Computer Vision (ICCV)* (2019).

[29] Kanazawa, A., Zhang, J. Y., Felsen, P. & Malik, J. Learning 3d human dynamics from video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

[30] Mehta, D. *et al.* XNect: Real-time multi-person 3D motion capture with a single RGB camera. In *ACM Transactions on Graphics* (2020).

[31] Rematas, K., Nguyen, C., Ritschel, T., Fritz, M. & Tuytelaars, T. Novel views of objects from a single image. In *arXiv* (2016).

[32] Rhodin, H., Constantin, V., Katircioglu, I., Salzmann, M. & Fua, P. Neural scene decomposition for multi-person motion capture. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

[33] Martinez, J., Hossain, R., Romero, J. & Little, J. J. A Simple Yet Effective Baseline for 3d Human Pose Estimation. In *IEEE International Conference on Computer Vision (ICCV)* (2017).

[34] Pavllo, D., Feichtenhofer, C., Grangier, D. & Auli, M. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

[35] Liu, J., Guang, Y. & Rojas, J. Gast-net: Graph attention spatio-temporal convolutional networks for 3d human pose estimation in video. In *arXiv* (2020).

[36] Cai, Y. *et al.* Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *IEEE International Conference on Computer Vision (ICCV)* (2019).

[37] Yiannakides, A., Aristidou, A. & Chrysanthou, Y. Real-time 3d human pose and motion reconstruction from monocular rgb videos. *Comput. Animat. Virtual Worlds* **30**, 1–12 (2019).

[38] Machado, A. S., Darmohray, D. M., Fayad, J., Marques, H. G. & Carey, M. R. A quantitative framework for whole-body coordination reveals specific deficits in freely walking ataxic mice. *Elife* **4**, e07892 (2015).

[39] Wei, S., Ramakrishna, V., Kanade, T. & Sheikh, Y. Convolutional pose machines. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[40] Card, G. & Dickinson, M. H. Visually mediated motor planning in the escape response of *Drosophila*. *Current Biology* **18**, 1300 – 1307 (2008).

[41] Wosnitza, A., Bockemühl, T., Dübbert, M., Scholz, H. & Büschges, A. Inter-leg coordination in the control of walking speed in *Drosophila*. *Journal of Experimental Biology* **216**, 480–491 (2013).

[42] Ionescu, C., Papava, D., Olaru, V. & Sminchisescu, C. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments (2014).

[43] De Bono, M. & Bargmann, C. I. Natural variation in a neuropeptide y receptor homolog modifies social behavior and food response in *C. elegans*. *Cell* **94**, 679–689 (1998).

[44] Budick, S. A. & O'Malley, D. M. Locomotor repertoire of the larval zebrafish: swimming, turning and prey capture. *Journal of Experimental Biology* **203**, 2565–2579 (2000).

[45] Louis, M., Huber, T., Benton, R., Sakmar, T. P. & Vosshall, L. B. Bilateral olfactory sensory input enhances chemotaxis behavior. *Nature Neuroscience* **11**, 187–199 (2008).

[46] Strauss, R. & Heisenberg, M. Coordination of legs during straight walking and turning in *Drosophila melanogaster*. *Journal of Comparative Physiology A* **167**, 403–412 (1990).

[47] Clarke, K. & Still, J. Gait analysis in the mouse. *Physiology & behavior* **66**, 723–729 (1999).

[48] Wiltschko, A. B. *et al.* Mapping sub-second structure in mouse behavior. *Neuron* **88**, 1121–1135 (2015).

[49] Hong, W. *et al.* Automated measurement of mouse social behaviors using depth sensing, video tracking, and machine learning. *Proceedings of the National Academy of Sciences* **112**, E5351–E5360 (2015).

[50] Mendes, C. S., Bartos, I., Akay, T., Márka, S. & Mann, R. S. Quantification of gait parameters in freely walking wild type and sensory deprived *Drosophila melanogaster*. *eLife* **2**, 231 (2013).

[51] Feng, K. *et al.* Distributed control of motor circuits for backward walking in *Drosophila* (2020).

[52] Alp Güler, R., Neverova, N. & Kokkinos, I. Densepose: Dense human pose estimation in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2018).

[53] Güler, R. A. & Kokkinos, I. Holopose: Holistic 3d human reconstruction in-the-wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

[54] Loper, M., Mahmood, N., Romero, J., Pons-Moll, G. & Black, M. J. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)* **34**, 248:1–248:16 (2015).

[55] Zhang, J. Y., Felsen, P., Kanazawa, A. & Malik, J. Predicting 3d human dynamics from video. In *IEEE International Conference on Computer Vision (ICCV)* (2019).

[56] Silvia Zuffi, T. B.-W. M. J. B., Angjoo Kanazawa. Three-d safari: Learning to estimate zebra pose, shape, and texture from images "in the wild". In *IEEE International Conferene on Computer Vision (ICCV)* (2019).

[57] Ioffe, S. & Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift **37**, 448–456 (2015).

[58] Nair, V. & Hinton, G. E. Rectified linear units improve restricted boltzmann machines. In *International Conference on Machine Learning (ICML)*, 807–814 (2010).

[59] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research* **15**, 1929–1958 (2014).

[60] He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[61] Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *arXiv* (2014).

[62] Sridhar, V. H., Roche, D. G. & Gingins, S. Tracktor: Image-based automated tracking of animal movement and behaviour. *Methods in Ecology and Evolution* **10**, 815–820 (2019).

[63] Günel, S., Rhodin, H. & Fua, P. What face and body shapes can tell us about height. In *IEEE International Conference on Computer Vision (ICCV) Workshops* (2019).