

Spatially recursive estimation and Gaussian process dynamic models of bat flapping flight

Matt Bender  · Li Tian · Xiaozhou Fan ·
Andrew Kurdila · Rolf Müller

Received: 19 July 2017 / Accepted: 11 September 2018 / Published online: 10 October 2018
© Springer Nature B.V. 2018

Abstract Bats exhibit exceptional agility, maneuverability, and efficiency during flight due to the complex articulated multibody structure of their wings and to the nonlinear and unsteady dynamics that govern their motion. While excellent progress has been made in the study of the kinematics of bat flapping flight, there still does not exist a dynamic model which is suitable for use in state estimation. This issue is typically overcome by using a few high-frame-rate cameras to capture motion; however, such systems are expensive and prone to measurement occlusion. This paper establishes a methodology that is designed to exploit an emerging class of experimental hardware which employs low-resolution, low-cost, and highly redundant imaging networks. The redundant camera network ameliorates the issue of self-occlusion, but the large-baseline, nonlinear motion of points in image space makes tracking difficult without a suitable motion prior. To remedy this issue, this paper exploits the tree topology of the bat skeleton and introduces a conditionally independent Bayes' filter implemented with inboard state correction. Our results show

that at low frame rates, this estimator performs better than both the standard and conditionally independent without inboard correction approaches for state estimation of an open kinematic chain. In addition to the estimation strategy, we construct a Gaussian process dynamic model (GPDM) of flight dynamics which we will use in future work as a suitable motion prior for state estimation. The GPDM presented in this paper is the first nonlinear dimensionality reduction of bat flight.

Keywords Motion capture · Bioinspired design · Gaussian process dynamic models · Model learning · State estimation

1 Introduction

A surge of research into micro-air vehicles (MAVs) has taken place over the past decade. The novelty of flapping-wing MAVs encourages optimism that significant performance enhancements may arise based on bioinspiration. Although substantial progress has been made in observing, modeling, and synthesizing flapping flight MAVs, most designs employ only a few degrees of freedom per wing.

These designs resemble those of an insect [51] or bird [31]. When compared to insects and birds, the potential benefits of using bats for MAV bioinspiration include enhanced maneuverability, stability, and efficiency. To date, researchers have synthesized 2

M. Bender (✉) · X. Fan · A. Kurdila · R. Müller
Mechanical Engineering, Virginia Tech, Blacksburg,
VA 24060, USA
e-mail: mattjb8@vt.edu

L. Tian
Electrical and Computer Engineering, Virginia Tech,
Blacksburg, VA, USA

R. Müller
VT-SDU International Lab, 27 Shanda South Road,
Jinan 250100, China

examples of bat-like robots. One incorporates a smart metal alloy (SMA) actuator [11], and another employs anisotropic membrane material to approximate the bat wing membrane [26, 27]. These examples are impressive, functional realizations that approximate bat flight. However, they do not approach the full skeletal complexity of bats. Because of the early stage of development of MAVs based on flapping flight, the need to develop rigorous biomimetic estimation, modeling, and design principles for such vehicles is pressing. Thus, the goal of this paper is to devise an experimental and analytical framework for the characterization of complex articulated flapping flight which can form the basis for rigorous biomimetic MAV design methodologies.

The main difficulty in realizing the full complexity of bat flight in flapping-wing MAVs is the fundamental lack of understanding of the kinematics and dynamics of the bat skeleton. While our kinematic definition is described in more detail in Sect. 5, it is characterized by a family of homogeneous transforms, \mathbf{H}_l^0 , which maps body fixed vectors on link l back to the inertial frame. These transforms are formulated as

$$\mathbf{H}_l^0 = \begin{bmatrix} \mathbf{R}_l^0 & \mathbf{r}_{0,l}^0 \\ \mathbf{0}^T & 1 \end{bmatrix},$$

where \mathbf{R}_l^0 is a rotation matrix which describes the relative orientation between the body fixed frame on link l and the inertial frame, and $\mathbf{r}_{0,l}^0$ is the location of the origin of the l^{th} body fixed frame expressed in the inertial frame. Both of these terms are functions of the joint coordinates $\mathbf{q}_{1:l}$ for all links $l \in \mathcal{T}$ where l is an address in the connectivity tree, \mathcal{T} , of the bat skeleton.

Using these kinematic variables, it is possible to express the equations governing the dynamics of bat flight in the form,

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \frac{\partial V}{\partial \mathbf{q}} = \boldsymbol{\tau}_a + \boldsymbol{\tau}_c \quad (1)$$

where \mathbf{M} is the mass matrix, \mathbf{C} is the Coriolis matrix, V is the system potential energy, $\boldsymbol{\tau}_a$ is a vector of aerodynamic forces and torques, $\boldsymbol{\tau}_c$ is a vector of control forces and torques, and $\mathbf{q} = [\mathbf{q}_1^T, \dots, \mathbf{q}_L^T]^T$ is the entire state vector. Despite the analytical formulation for the nonlinear mass matrix $\mathbf{M}(\mathbf{q})$ we cannot determine values for link lengths, masses, or inertias with enough certainty which would render it useful for state prediction. Additionally, the Coriolis matrix $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ depends on the an-isotropic, internally actuated wing membrane for which only preliminary investigations have

been conducted [10]. Furthermore, the complex aerodynamic torques, $\boldsymbol{\tau}_a$, are caused in part by the highly articulated wing structure of bats. The study of this class of fluid flows still poses numerous open research challenges in the field of computational fluid dynamics [32] because unsteady, nonlinear effects are substantial and cannot be neglected [14]. Performing online computation of these aerodynamics is intractable because they require on the order of 10^6 – 10^8 computational degrees of freedom [44, 46] and, consequently, hours or days to complete [45]. Finally, the control forces and torques, $\boldsymbol{\tau}_c$, are unknown and unmeasurable with current data acquisition hardware. Thus, even if all the previous terms were known, we still do not know how to drive the system. Thus, Eq. 1 cannot be used for robotic design, prediction in recursive filtering algorithms, or dimensionality reduction.

It must be emphasized that identifying joint states and identifying the dynamics in Eq. 1 are inherently coupled. Recall that any recursive probabilistic filter includes a prediction step which generates a preliminary state estimate from the equations of dynamics. We have already noted, however, that the classical equation of robotics in Eq. 1 is poorly understood rendering them effectively useless in this context. Conversely, algorithms which approximate the equations of robotics in Eq. 1 can be constructed by relying on experimentally determined kinematic variables, $\{\mathbf{q}_l\}_{l=1}^L$, but we do not have such joint trajectories—estimating them is the purpose of this paper. After estimating joint trajectories, we learn a Gaussian process dynamic model (GPDM) which defines a discrete evolution law within a latent space and a mapping from latent space to the full state space. To the best of our knowledge, this is the first nonlinear dimensionality reduction of bat flight.

To begin our detailed discussion of our method, Sect. 2 provides a literature review of related research in bat flight and state-of-the-art estimation methods. In Sect. 3, we present an overview of our framework for estimating joint states without a suitable motion model. Next, Sect. 4 discusses the theoretical formulation for our estimator and reviews the GPDM formulation. Section 5 presents synthetic simulations and experimental results which show that our estimator outperforms traditional methods and that the manifold we identify using GPDM is consistent with the observed motions. Finally, Sect. 6 provides concluding remarks and opportunities for future work.

2 Related research

2.1 Bat flight studies

The kinematics and aerodynamics of bat flapping flight is exceptionally rich and complex and has been studied by many researchers to date. Previous studies of bat flight have employed a small number of high-frame-rate, high-resolution, low-distortion cameras [7, 10, 18, 28–30, 42, 50]. Due to the quality of cameras used, the motion of points in the image space, while nonlinear, has a small baseline in these studies. This allows for a direct application of standard techniques like optical flow and Lucas–Kanade tracking [7] to be used to establish temporal marker correspondence. Because of the small number of cameras, initialization of marker points in each frame can be done manually with little or modest user intervention. Software packages exist for processing this data which are open source [8, 16] or commercially available [40].

While the experimental hardware used in these studies allows for the use of more or less standard tracking and correspondence algorithms, the limited number of cameras is prone to systematic, pervasive, and periodic self-occlusions [30, 42]. In many papers this effect is substantial. In fact, the loss of inertial estimates due to self-occlusion is reported to occur during 15.6% of the observation time during an experiment [30]. Additionally, many of the previous studies only use data from a single wing for dimensionality reduction [30], CFD studies [44–46], and PIV measurements [29, 30, 50].

Finally, the previous experiments were conducted with fruit-eating bats. While these bats have highly articulated wings, their food does not evade them. Bats that are predators of insects must pursue and capture prey, often in dense forest environments. Thus, they typically exhibit more agile maneuvers during capture of prey [3] and, therefore, present a formidable challenge to motion studies. Table 1 summarizes the experimental paradigms used in previous studies and provides a contrast to the experimental methods presented in this paper.

As shown in Table 1, the experimental facility developed in this paper uses an order of magnitude more cameras, operating at $\frac{1}{2}$ – $\frac{1}{8}$ the frame rate, and at a substantially lower resolution than that used historically. The cameras used in these experiments are GoPro Hero 3+ Black cameras which record 720p video at 120 frames per second (fps). These cameras were selected because they provide adequate sampling rate and resolution at an order of magnitude lower cost than typical high-speed, high-resolution cameras. An example of the images produced by the system is shown in Fig. 1. In the figure, a top-view camera can see the back of the bat and tops of the wings, and a side-facing camera can see the top of the right wing in some frames and the bottom of the right wing in others. In future work, we will study and compare the complex maneuverability exhibited by an assortment of species of bats. Such maneuvers will require the highly redundant camera system developed for our experiments to minimize or eliminate the problem of self-occlusion.

Table 1 Bat flight motion capture system specifications

Source	Cams	fps	Res	Points	Species
[1]	2	200	Film	6	<i>R. ferrumequinum</i>
[42]	2	500	Unknown	≈ 20	<i>C. brachyotis</i>
[18]	4	200	1024	17	<i>C. brachyotis</i>
[50]	2	250	Unknown	9	<i>G. soricina</i>
[7, 10, 28–30]	3–4	1000	1024	17–32	<i>C. brachyotis</i> <i>R. aegyptiacus</i> <i>P. pumilus</i> <i>E. hypomelanus</i> <i>P. vampyrus</i>
Our array (this paper)	30 (17)	120	720p	≈ 120 (34)	<i>R. ferrumequinum</i> <i>H. pratti</i> (<i>H. armiger</i>)



Fig. 1 Point occlusions are frequent and periodic in bat flight. The top camera captures the downstroke well, but occlusion occurs on the upstroke. Conversely, the side camera captures the upstroke well, but portions of the downstroke are occluded

While the experimental hardware used in our experiments will eliminate the self-occlusion problem in many flight regimes and allows motion capture of both wings, the low-cost cameras introduce difficulties which are not observed in conventional studies that employ a few high-quality cameras, such as lens distortion, low-resolution, and large-baseline motion. These difficulties cause the standard methods such as feature descriptor matching or optical flow to fail when applied to our image sets. Thus, we look to recent advancements in articulated motion estimation for approaches which can be adapted to our application.

2.2 Estimation of Kinematic Pose

Although standard tracking and correspondence methods often fail for the data sets used in this paper, many researchers have studied kinematic estimation of biological systems, and alternative methods have been proposed. The three-dimensional reconstruction of biological kinematics has been studied for a long time, and many survey articles on the topic exist. One popular review [25] discusses the state of the art in human motion capture techniques for monocular, stereo, and redundant camera systems. Another review [13] focuses on motion estimation techniques for the human hand, including the use of various numbers of

cameras for pose determination. A more recent review [21] summarizes techniques for estimating 2D motion of characters in monocular image sequences (using only one camera). These reviews are only a few of those that exist, and the reader is directed to the references of these reviews for a more exhaustive list of common state estimation techniques for motion capture studies.

Within these reviews of articulated motion capture, many themes emerge in the approaches developed to date. All of the current kinematic estimation algorithms use either deterministic or probabilistic methods. Deterministic methods aim to identify pose via constrained optimization. Conversely, probabilistic methods aim to model the subject's motion and the sensors with which the object is observed to iteratively predict and correct the estimated state of the system. Additionally, certain variants of the stochastic motion capture approach are capable of representing multiple hypotheses, which makes the approaches attractive when ambiguities in pose exist.

While these methods differ in execution, both deterministic and probabilistic formulations of the problem can be applied as a batch process or as a recursive algorithm. Batch solutions use data from all time steps and all sensors to estimate all states at all time steps simultaneously. Due to the dimensionality of the biological kinematic estimation problem, this approach is generally computationally demanding or intractable altogether. Thus, a common formulation is to estimate all states at a single time given all sensor information from that time step. This is the most common paradigm in motion estimation and has been employed in previous studies of bat flight [4,6,7]. Recursive solutions use a subset of the collected data to estimate a portion of the pose states at a block of the time steps. Because our formation is probabilistic and recursive, the remainder of the review will focus on such methods.

Probabilistic estimation approaches approximate a target posterior probability distribution by making assumptions about the general form of the distribution, the prediction of states from one time step to the next, and correction of estimates using sensor information [41]. One common method for determining pose from sets of images is to predict all states in the kinematic chain simultaneously. This approach usually requires that spatial and temporal correspondence is performed a priori within each camera view. Previous work conducted by the authors has formulated

the motion estimation problem in this fashion [4]. Some work has been done by bootstrapping optical flow or expectation maximization to solve the correspondence problem online [12]. For optical flow to be applicable, small baseline motion is desirable, and in some instances required. Due to the low frame rate used for our experiments, this is not the case in our data.

Another method for approximating the posterior is to partition the state space into a hierarchical tree and represent the distribution as piecewise constant over that tree [35, 36, 39]. Typically the process of partitioning is conducted using a hierarchical search. The initial state partitions are coarse, and the probability density function is typically evaluated at the center of each bin. Bins with sufficient empirical probability density are further partitioned. The posterior distribution is approximated as the piecewise constant combination of the finest partitions. While this method could potentially be applied to our experiments, the computational cost increases as the bin refinement increases. For our purposes, we would need very fine bin partitions to achieve reconstruction results which are accurate enough for use in CFD simulations and dimensionality reduction. Thus, this method is not ideal for our experiments. In another work [17], the tree structure is used for hierarchical matching in two-dimensional space to estimate the posterior of the three-dimensional pose. Poses are then refined by projecting a three-dimensional estimate into each camera, and K-best trajectory tracks are chosen using a Viterbi algorithm. Finally, refinement is done in image space.

More recent works use probabilistic graphical models (GMs) to represent dependencies between states and sensor information. The GM methods can employ directed or undirected graphs and can be formulated using a priori knowledge of the structure of the state space or learn the relationship from training data. A recent work represents kinematic constraints stochastically as connections between nodes in the graph [33]. Each node represents the states which describe the orientation of a body with respect to the surrounding bodies. This work constructs the posterior as a product of Bayes' rule on individual parts in which body parts are dependent on each other. Training data is required to learn the relationship between parts of the model. Again, these methods are also not applicable to our problem because training data, which does not exist for the bat, is required for implementation.

Another approach which exploits the tree structure of the kinematic skeleton is to partition the state space and perform estimation on each partition sequentially. This approach would assume that groups of states are conditionally independent from one another. The closest estimation strategy which appears in the literature is partitioned sampling [23] which is an extension of ICondensation [19] as used for tracking hand motion. The algorithm partitions the state space into a tree structure assuming that the motion of the palm can be estimated independently of the fingers. Using the corrected palm state, the algorithm works outward through the connectivity tree to estimate the position of the fingers. This approach employs conditional independence (CI) but differs from ours in that the algorithm is implemented using a particle filter and the palm state is not re-corrected when the finger states are estimated. Our estimation framework also assumes conditional independence but is implemented using an unscented Kalman filter and applies inboard state correction (ISC) as new state partitions are estimated. This feature, as demonstrated in this paper, helps prevent over-fitting noise in inboard link states which dramatically reduces the amount of error observed in outboard link states.

2.3 Estimation of nonlinear dynamic models

The second goal of this paper is to develop a suitable low-order representation of bat flight which can be used for automated tracking in future studies. While nonlinear dimensionality reduction of bat flight has not been performed previously, such analysis is pervasive in human motion studies [20, 43, 47, 49]. These low-order representations of human motion have been used to automate tracking in commercial motion capture systems, drive pose synthesis for animations, and plan joint trajectories for humanoid robots. This paper develops an approach which is based on the methods presented in [43, 47, 49] which develop Gaussian process dynamic models (GDPM) for learning low-order representations of human motions. The GDPM framework extends the work in [20], which introduces Gaussian process latent variable models (GPLVM), by preserving the continuity of training data. This is important when little training data is available—as is the case for our problem. The approach derived in this paper is described in more detail in Sect. 4.3.

3 Overview of our method

In this paper we exploit the connectivity of the multi-body model of the bat bioskeletal structure to develop a conditionally independent spatially and temporally recursive Bayes' filter which incorporates inboard state correction (ISC) for estimating bat motion from video sequences. The essential feature of this formulation is that all previously corrected states are re-corrected when each new partition of the state space is corrected. Figure 2 visually summarizes the sequence of steps in the conditionally independent Bayesian estimation procedure which incorporates inboard state correction introduced in this paper.

The estimation procedure progresses from the upper left to upper right in the top row of panels and then to the next row from lower left to lower right. The state estimate for links 1–4 at time step $k - 1$ is depicted in the upper left panel. In the next frame, the bat has moved forward, but the state estimate remains the same due to the random-walk motion model. In the next panel to the right, measurements associated with the base body are used to correct the estimate of link 1. In the rightmost, top row panel, the updated estimate of link 1 state is used in the correction step for link 2. It is important to note in this panel that the estimates are much closer to the observations even though the observations of only the root link have been incorporated. In the succeeding panels, the algorithm traverses to *conditionally independent* collections of states and their associated measurements. The recursion proceeds from the root of the connectivity tree to its outermost leaves. The novelty in our recursion is that lower link states are corrected as new link states are estimated. For example, when link 3 states are corrected, the states for links 1 and 2 are re-corrected to insure that the initial correction steps did not over fit noise. While re-correcting previously estimated joint states requires additional computational resources, we demonstrate through a collection of representative studies that in comparison with both the standard method which corrects all states simultaneously and the conditionally independent (CI) method which partitions the state space but does not re-correct link states as new states are estimated, our inboard state correction (ISC) methodology yields far better estimation results. Our discussion of the proposed methodology begins in Sect. 4 where we present our kinematic representation of bat flight, the derivation of the conditionally independent spatially recursive Bayes' fil-

ter with ISC, and a review of the foundational theory of Gaussian process dynamic models for identifying a low-order, nonlinear model of bat flapping flight.

4 Theoretical formulation

In their full complexity, bats are infinite-dimensional, nonlinear systems because of their elaborate musculoskeletal system and flexible wing membrane. However, infinite-dimensional models are intractable for system identification and state estimation, so some approximation is required. Approaches to this problem can be organized into rigid and nonrigid models: this paper is concerned with the motion of the bones modeled as rigid links connected by ideal joints. This section defines the kinematics and dynamics of the skeleton and presents our conditionally independent estimation algorithm.

4.1 Kinematics and dynamics of bat flight

To develop a suitable kinematic model, we assume that the bat skeleton is well-approximated by an open kinematic chain. We assume that the base body is comprised of three links: one zero-length link which accounts for $x - y - z$ translation, and two rigid links joined by a rotational degree of freedom at the base of the tail. This rotational DOF is added to enable the shoulder markers to move with respect to one another. This choice is justified because the markers cannot be placed at the center of rotation of the shoulder. They are instead placed on the shoulder blades which move laterally as the bat flaps its wings. This relative motion between points can cause large estimation errors if the body is assumed to be a single rigid link. The wings of the bat are attached to the ends of the two base body links and are comprised of a humerus, a radius, three metacarpals, three first phalanges, and three second phalanges. While the bat actually has 5 digits on each hand, the claw digit is very short and does not affect the overall wing motion. Similarly, the second digit is very close to the third digit and only consists of a metacarpal. Therefore, both of these links are omitted from our analysis. We assume that the shoulder joints have three DOF, the elbow joints have one DOF, and the metacarpal and phalangeal joints have two DOF. According to this definition, the complete state space for the bat has 51 DOF. All gener-

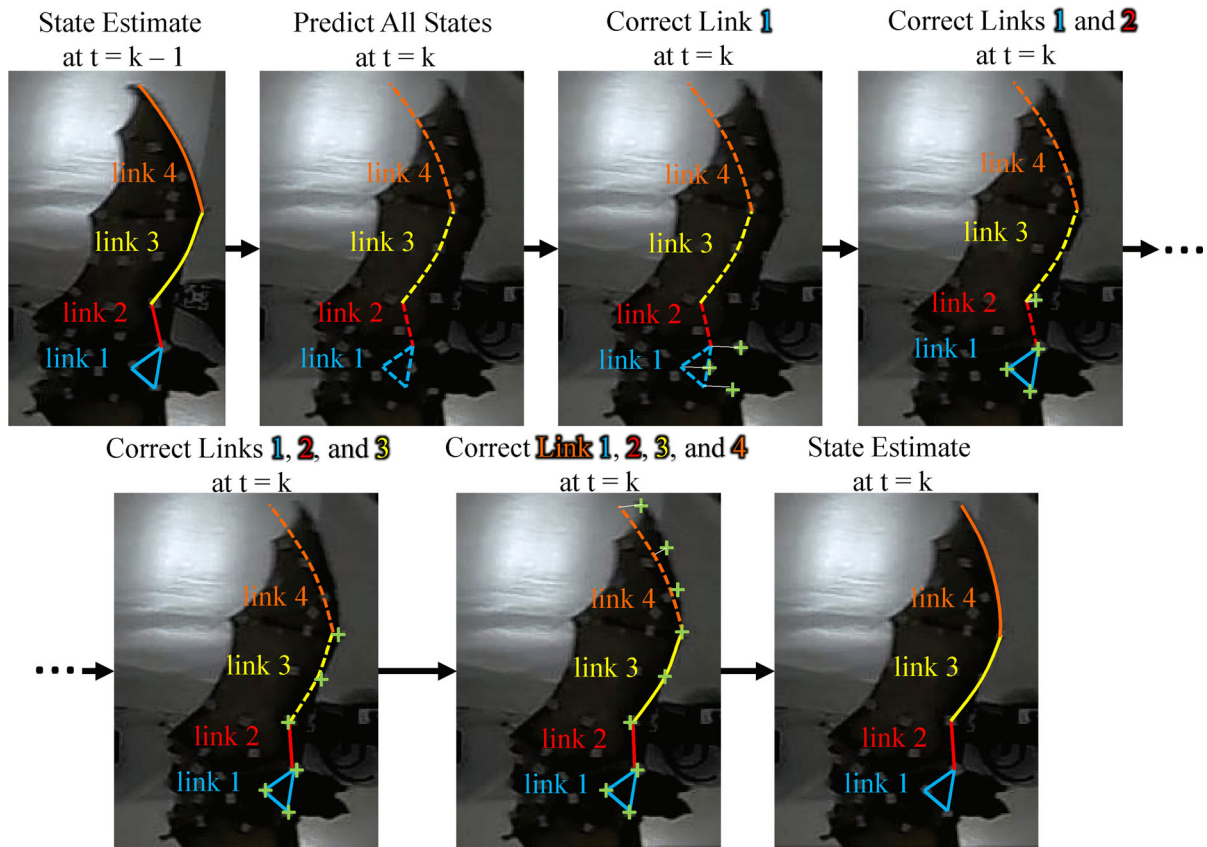


Fig. 2 Conditionally independent spatially recursive Bayesian estimation. Unlike the i.i.d. approach to recursive estimation, the conditionally independent estimator corrects all previous link

states as new link states are added. This prevents over-fitting noise for root links which improves estimation accuracy

alized joint coordinates are parameterized using the conventional and well-known DH convention [34]. A detailed discussion of the kinematics can be found in [5].

From this kinematic definition, we can assemble state vectors for individual links and groups of links. The state vector for the l th link, \mathbf{q}_l , contains all degrees of freedom for that link. These state vectors can be grouped using the notation $\mathbf{q}_{a:b}$, where b is a link index which is outboard with respect to link index a . While we discuss the sensor model and the mapping from states, \mathbf{q} , to measurements, ϕ , in more detail below, the measurements of features in image space for links 1 to l over time steps 1 to k are denoted as $\phi_{1:l,1:k}$. This notation is critical in the derivation of the conditionally independent Bayes' filter with inboard state correction (ISC).

4.2 Conditionally independent Bayes' filter with inboard state correction

We begin the derivation of the filter by applying the chain rule to the global posterior distribution

$$p(\mathbf{q}_{1:l,k} | \phi_{1:l,1:k}) = p(\mathbf{q}_{l,k} | \mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k}) * p(\mathbf{q}_{1:l-1,k} | \phi_{1:l,1:k}).$$

where $\mathbf{q}_{l,k}$ is the state vector for link l , and $\phi_{1:l,1:k}$ are the camera measurements of links 1 to l for time steps 1 to k . Using the chain rule on the global posterior, $p(\mathbf{q}_{1:l,k} | \phi_{1:l,1:k})$, allows us to isolate the probability of the state vector of link l given the states of the previous links, $\mathbf{q}_{1:l-1,k}$, and all the measurements.

To simply this distribution, the first conditional independence property that we apply assumes that the states of links 1 to $l - 1$ are independent of measurements of link l given measurements of links 1 to $l - 1$. We express this fact using the notation, $\mathbf{q}_{1:l-1,k} \perp\!\!\!\perp$

$\phi_{l,1:k}|\phi_{1:l-1,1:k}$, and we can write

$$p(\mathbf{q}_{1:l,k}|\phi_{1:l,1:k}) = p(\mathbf{q}_{l,k}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k}) * p(\mathbf{q}_{1:l-1,k}|\phi_{1:l-1,1:k}).$$

We can repeatedly apply the chain rule to $p(\mathbf{q}_{1:l-1,k}|\phi_{1:l-1,1:k})$ in Eq. 2 until it follows that

$$p(\mathbf{q}_{1:l,k}|\phi_{1:l,1:k}) = \left(\prod_{l=2}^L p(\mathbf{q}_{l,k}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k}) \right) * p(\mathbf{q}_{1,k}|\phi_{1,1:k}). \quad (2)$$

Note that the second term on the right-hand side is the posterior distribution of the joint variables of link 1 given the sensor measurements of that link. This posterior can be estimated using any implementation of the Kalman filter—we choose the unscented Kalman filter. Thus, we need to formulate an expression for $p(\mathbf{q}_{l,k}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k})$ to complete the spatially recursive filter derivation.

Applying Bayes' rule to $p(\mathbf{q}_{l,k}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k})$, we have

$$p(\mathbf{q}_{l,k}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k}) = \eta p(\phi_{1:l,k}|\mathbf{q}_{1:l,k}; \phi_{1:l,1:k-1}) * p(\mathbf{q}_{l,k}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k-1}).$$

Additionally, current measurements are independent of past measurements given the current state which is stated as, $\phi_{1:l,k} \perp\!\!\!\perp \phi_{1:l,1:k-1}|\mathbf{q}_{1:l,k}$. Applying this conditional independence condition, we get

$$p(\mathbf{q}_{l,k}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k}) = \eta p(\phi_{1:l,k}|\mathbf{q}_{1:l,k}) * p(\mathbf{q}_{l,k}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k-1}). \quad (3)$$

Expanding $p(\mathbf{q}_{l,k}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k-1})$ using the definition of the marginal probability, we have

$$\begin{aligned} & p(\mathbf{q}_{l,k}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k-1}) \\ &= \int p(\mathbf{q}_{l,k}|\mathbf{q}_{l,k-1}; \phi_{1:l,1:k-1}; \mathbf{q}_{1:l-1,k}) \\ & \quad * p(\mathbf{q}_{l,k-1}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k-1}) d\mathbf{q}_{l,k-1}. \end{aligned}$$

Finally, we can apply conditional independence conditions to simplify this expression. As we discuss in more detail in the next section, a random walk motion model is assumed for prediction of the states during kinematic estimation. Because of this motion model and the tree structure of the bat skeleton, we can assume that joint coordinates of link l at time k are independent of all other joint states at time k given the previous state of link l . That is, we have $\mathbf{q}_{l,k} \perp\!\!\!\perp \mathbf{q}_{1:l-1,k}|\mathbf{q}_{l,k-1}$.

Also, the joint coordinates at time k are independent of previous measurements given the previous state, $\mathbf{q}_{l,k} \perp\!\!\!\perp \phi_{1:l,1:k-1}|\mathbf{q}_{l,k-1}$. Finally, a past link state is independent of the future link state given current measurement information, $\mathbf{q}_{l,k-1} \perp\!\!\!\perp \mathbf{q}_{1:l-1,k}|\phi_{1:l,1:k-1}$. So, we conclude that

$$p(\mathbf{q}_{l,k}|\mathbf{q}_{1:l-1,k}; \phi_{1:l,1:k-1}) = \int p(\mathbf{q}_{l,k}|\mathbf{q}_{l,k-1}) * p(\mathbf{q}_{l,k-1}|\phi_{1:l,1:k-1}) d\mathbf{q}_{l,k-1}. \quad (4)$$

The recursive Bayes' filter can be summarized by Eqs. 2–4. Equation 2 computes the joint probability of all links up to and including link l . Equation 3 incorporates new sensor information as well as sensor information from all previously corrected inboard link states. Finally, the prediction is computed using Eq. 4.

The Bayes' filter can be propagated using any probabilistic filter of choice, such as Kalman or particle filters. Particle filters are often applied to estimation problems which expect multi-modal distributions which are common in single-frame human pose estimation studies. This paper however performs temporal tracking of bats using a highly redundant camera system, so pose ambiguity is not an issue and multi-modal belief propagation is not necessary. Thus, we choose the unscented Kalman filter which can accommodate the nonlinear skeletal kinematics of our problem and does not require analytical Jacobian matrices for the motion and sensor models—the motion model is linear, but the sensor model is highly nonlinear.

In the derivation of the recursive Bayes' filter we assume that all DOFs for each link are adequately constrained by the sensor data available. However, this is not always true in bat motion capture studies. For example, the humerus has three degrees of freedom, but the length to diameter ratio of this link is such that any collection of markers are insufficient to identify all its degrees of freedom. Thus, we group this link with the radius and effectively estimate the planar orientation and geometry of the entire arm simultaneously. Additionally, we group the metacarpals together, first phalanges together, and second phalanges together for a total of 5 groups. While we group links in this manner, our derivation of the spatially recursive Bayes' filter holds for any group assignment so long as inboard links are estimated before or at the same time as outboard links. The next section presents the motion model used for state prediction.

The absence of sufficiently accurate motion models for biologically articulated motion permeates the motion capture literature [25], and many use random walk for state propagation [4, 9, 15]. Thus, we use this model as well,

$$\mathbf{q}_{1:l,k+1} = \mathbf{q}_{1:l,k} + \mathbf{w},$$

where \mathbf{w} is zero-mean Gaussian noise with constant covariance. This motion model does not predict the direction of motion and assumes that the state does not change dramatically between adjacent time steps. The uncertainty injected at each prediction step can be estimated from inspection of the experimental videos. The bat's body is roughly 100 mm long from the tip of the ears to the end of the feet, and it travels about one body length in 7 frames. Thus, the zero-mean uncertainty in body translation is 15 mm. For angular uncertainty, we assume that the humerus undergoes the largest rotation in order to flap the wing up and down. We also assume that the entire upstroke is roughly 90° of motion, which occurs over approximately 8 frames. From these observations, we estimate that the wing joint standard deviation is approximately 12°. Despite these large uncertainties induced by the poor understanding of bat motion, the estimator benefits from a very accurate representation of the sensor model used for correction.

Although our cameras are low cost and low resolution, camera imaging models are well known [22, 38]. While GoPro camera lenses possess a large amount of nonlinear distortion, we assume that this distortion can be removed prior to estimation. With distortion removed we assume that the camera is modeled by the linear projection model in [22]

$$\phi = \begin{Bmatrix} \psi_x \\ \psi_y \end{Bmatrix} = \frac{1}{c\lambda} {}^c\mathbf{K}\mathbf{\Pi}_0\mathbf{H}_0^c\mathbf{x}^0, \quad (5)$$

where λ is distance from the camera focal center to the point \mathbf{x}^0 in inertial coordinates, ${}^c\mathbf{K}$ is the camera calibration matrix for the c th camera, $\mathbf{\Pi}_0$ is the canonical projection matrix, and \mathbf{H}_0^c is the homogeneous transform from the inertial basis to the camera basis. While this camera model maps a point in the inertial basis to a camera measurement, the motion model is in joint space. We map between joint space and Euclidean space using the nonlinear kinematics defined in Sect. 5. Thus, the entire sensor model becomes

$$\phi = \begin{Bmatrix} \psi_x \\ \psi_y \end{Bmatrix} = \frac{1}{c\lambda} {}^c\mathbf{K}\mathbf{\Pi}_0\mathbf{H}_0^c\mathbf{H}_l^0(\mathbf{q}_{1:l})\mathbf{x}^l + \mathbf{v}, \quad (6)$$

where ϕ is the location of a feature in the image space of camera c , $\mathbf{H}_l^0(\mathbf{q}_{1:l})\mathbf{x}^l$ maps a point from the body fixed vector \mathbf{x}^l to the inertial frame, and \mathbf{v} is Gaussian noise.

One source of error in our sensor measurements is due to uncertainty in the calibration parameters calculated by the MultiCamSelfCal [37] toolbox. The reprojection error reported by the toolbox is consistently 1.5 pixels mean with 1.5 pixels standard deviation. Thus, in our sensor model we assume 3 pixels of measurement uncertainty.

A previous effort by the authors [4] presented results using the standard UKF algorithm, but here we employ the derived algorithm that is recursive in space and in time. This algorithm is demonstrated on both synthetic and experimental data in Sect. 5. Estimation accuracy is compared to the standard UKF estimator and the CI estimator without inboard correction. Our estimator is capable of producing adequate reconstructions of bat flight; however, we require manual correspondence of feature points in images due to large-baseline motion and a poor motion model. Thus, the second goal of this paper is to identify a low-order model of bat flight which can be used for the prediction step in the future.

4.3 Gaussian process dynamic models

This study uses GPDM to determine a low-order, nonlinear dynamic model from the kinematic states estimated by our ISC unscented Kalman filter. For implementation details, refer to the code provided in [48]. We provide a review of the foundational theory here which was originally derived for dimensionality reduction of human motion [43, 47, 49].

Given a data sequence $\mathbf{Q} := \{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_K\}$ which contains K observations of the N -dimensional state vector $\mathbf{q}_k \in \mathbb{R}^N \quad \forall k \in [1, 2, \dots, K]$, we define the latent space dynamics and the mapping from latent space to kinematic pose space as

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{A}) + \mathbf{n}_x, \quad (7)$$

$$\mathbf{q}_k = \mathbf{g}(\mathbf{x}_k, \mathbf{B}) + \mathbf{n}_y, \quad (8)$$

where $\mathbf{x}_k \in \mathbb{R}^D$ is a D -dimensional latent space in which the dynamics are embedded, \mathbf{n}_x is process noise, and \mathbf{n}_y is projection noise. The dynamic model, \mathbf{f} , and the mapping from latent space to feature space, \mathbf{g} , are defined as

$$\mathbf{f}(\mathbf{x}, \mathbf{A}) = \sum_j \mathbf{a}_j \phi(\mathbf{x}_j, \mathbf{x}) = \mathbf{A} \boldsymbol{\phi}(\mathbf{x}), \quad (9)$$

$$\mathbf{g}(\mathbf{x}, \mathbf{B}) = \sum_j \mathbf{b}_j \psi(\mathbf{x}_j, \mathbf{x}) = \mathbf{B} \boldsymbol{\psi}(\mathbf{x}), \quad (10)$$

where the vector-valued weighting coefficients, \mathbf{a}_j and \mathbf{b}_j , can be assembled into matrices such that $\mathbf{A} := [\mathbf{a}_1, \mathbf{a}_2, \dots]$ and $\mathbf{B} := [\mathbf{b}_1, \mathbf{b}_2, \dots]$. Thus, evaluating \mathbf{f} and \mathbf{g} can be written as a linear combination of basis functions and coefficient matrices, where $\boldsymbol{\phi}(\mathbf{x}) = [\phi(\mathbf{x}_1, \mathbf{x}), \phi(\mathbf{x}_2, \mathbf{x}), \dots]^T$, and $\boldsymbol{\psi}(\mathbf{x}) = [\psi(\mathbf{x}_1, \mathbf{x}), \psi(\mathbf{x}_2, \mathbf{x}), \dots]^T$. The individual basis functions are defined as,

$$\begin{aligned} \phi(\mathbf{x}_j, \mathbf{x}) &= \alpha_1 \exp\left(-\frac{\alpha_2}{2} \|\mathbf{x} - \mathbf{x}_j\|^2\right) \\ &\quad + \alpha_3 \mathbf{x}^T \mathbf{x}_j + \alpha_4^{-1} \delta(\mathbf{x}, \mathbf{x}_j), \end{aligned} \quad (11)$$

$$\begin{aligned} \psi(\mathbf{x}_j, \mathbf{x}) &= \beta_1 \exp\left(-\frac{\beta_2}{2} \|\mathbf{x} - \mathbf{x}_j\|^2\right) \\ &\quad + \beta_3^{-1} \delta(\mathbf{x}, \mathbf{x}_j), \end{aligned} \quad (12)$$

where $\boldsymbol{\alpha} := [\alpha_1, \alpha_2, \alpha_3, \alpha_4]$ and $\boldsymbol{\beta} := [\beta_1, \beta_2, \beta_3]$ are vectors of hyper-parameters which control the shape of each basis function. These model parameters are optimized to fit the given data set.

One method for fitting this model to a data set is to choose a fixed set of hyper-parameters and then optimize over \mathbf{A} and \mathbf{B} [2]. However, this approach can create a model which is specific to a particular trial or specimen. To identify a model which is more general, authors in [43, 47, 49] marginalize over the parameters in \mathbf{A} and \mathbf{B} and optimize latent variables and hyper-parameters. To perform this optimization, we formulate the joint probability of the collected data \mathbf{Q} , the set of all latent variables $\mathbf{X} := [\mathbf{x}_1, \dots, \mathbf{x}_K]$, the vectors of hyper-parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, and a weighting matrix \mathbf{W} as

$$\begin{aligned} p(\mathbf{Q}, \mathbf{X}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{W}) \\ = p(\mathbf{Q}|\mathbf{X}, \boldsymbol{\beta}, \mathbf{W}) p(\mathbf{X}|\boldsymbol{\alpha}) p(\boldsymbol{\alpha}) p(\boldsymbol{\beta}) p(\mathbf{W}). \end{aligned} \quad (13)$$

The first term on the right quantifies the probability of the mapping from latent space to feature space. An analytical expression for this probability is obtained by placing a Gaussian prior on the columns of \mathbf{B} and marginalizing over \mathbf{g} . This eliminates the dependency of the model on the parameters, and only hyper-parameters and latent states must be identified. The

marginalization can be done in closed form [24] to obtain

$$\begin{aligned} p(\mathbf{Q}|\mathbf{X}, \boldsymbol{\beta}, \mathbf{W}) &= \frac{|\mathbf{W}|^K}{\sqrt{(2\pi)^{KN} |\mathbf{K}_q|^N}} * \\ &\quad \times \exp\left(-\frac{1}{2} \text{trace}(\mathbf{K}_q \mathbf{Q} \mathbf{W}^2 \mathbf{Q}^T)\right), \end{aligned} \quad (14)$$

where $\mathbf{W} := \text{diag}(w_1, w_2, \dots, w_N)$ weights \mathbf{Q} so that the model does not over fit degrees of freedom with large variations and ignore degrees of freedom with small variations. The covariance kernel, \mathbf{K}_q , is used to quantify the closeness of points in the latent space. Each element in this matrix uses the kernel in Eq. 12, and the matrix is assembled as

$$(\mathbf{K}_q)_{ij} = \beta_1 \exp\left(-\frac{\beta_2}{2} \|\mathbf{x}_i - \mathbf{x}_j\|^2\right) + \beta_3^{-1} \delta(\mathbf{x}_i, \mathbf{x}_j), \quad (15)$$

where i indexes over the number of input data points, and j indexes over the number of basis function centers. The number of input points and the number of basis function centers are not required to be equal. However, in this implementation, they are equal.

The second term on the right-hand side of Eq. 13 quantifies the probability of a given set of dynamics in the latent space. Again, we assume a Gaussian prior on the columns of \mathbf{A} such that they can be marginalized out of the equation. This probability is formulated as

$$p(\mathbf{X}|\boldsymbol{\alpha}) = \int p(\mathbf{X}|\mathbf{A}, \boldsymbol{\alpha}) p(\mathbf{A}|\boldsymbol{\alpha}) d\mathbf{A}. \quad (16)$$

The final distribution is independent of the model parameters, \mathbf{A} , and the dynamic model only depends on hyper-parameters, $\boldsymbol{\alpha}$, and the latent states, \mathbf{X} . Furthermore, by assuming the latent space dynamics are Markovian, this probability can be computed in closed form as

$$\begin{aligned} p(\mathbf{X}|\boldsymbol{\alpha}) &= \frac{p(\mathbf{x}_1)}{\sqrt{(2\pi)^{(K-1)D} |\mathbf{K}_x|^D}} * \\ &\quad \exp\left(-\frac{1}{2} \text{trace}(\mathbf{K}_x^{-1} \mathbf{X}_{2:K} \mathbf{X}_{2:K}^T)\right). \end{aligned} \quad (17)$$

In this equation, the latent variables are nondimensional and do not require a weighting matrix. Additionally, the matrix $\mathbf{X}_{2:K}$ only contains latent states from time steps

2 to K . Finally, the covariance matrix uses the kernel function from Eq. 11 and is assembled as

$$(\mathbf{K}_x)_{ij} = \alpha_1 \exp\left(-\frac{\alpha_2}{2} \|\mathbf{x}_i - \mathbf{x}_j\|^2\right) + \alpha_3 \mathbf{x}_i^T \mathbf{x}_j + \alpha_4^{-1} \delta(\mathbf{x}_i, \mathbf{x}_j), \quad (18)$$

for all $i, j \in [2, \dots, K]$. The next terms in the distribution in Eq. 13 are the priors on α , β which are formulated as,

$$p(\alpha) = \prod_i \alpha_i^{-1} \quad (19)$$

$$p(\beta) = \prod_i \beta_i^{-1} \quad (20)$$

These priors serve to constrain the hyper-parameters to small values.

The final term in Eq. 13 is the prior on the weighting matrix. For this paper, a half normal prior is assumed on each entry in \mathbf{W}

$$p(\mathbf{W}) = \prod_{m=1}^D \frac{2}{\kappa \sqrt{2\pi}} \exp\left(-\frac{w_m^2}{2\kappa^2}\right), \quad (21)$$

where κ is a scaling parameter set to 1000 in our experiments and w_m is greater than 0. This prior insures that the weights do not introduce singularities in our estimation and that dimensions with small variations do not dominate the optimization [49].

The probability distribution in Eq. 13 can be maximized by minimizing the negative log likelihood of the distribution which is expressed as

$$\begin{aligned} L = & \frac{D}{2} \ln \|\mathbf{K}_x\| + \frac{N}{2} \ln \|\mathbf{K}_q\| - K \ln |\mathbf{W}| \\ & + \frac{1}{2} \text{trace}\left(\mathbf{K}_x^{-1} \mathbf{X}_{2:K} \mathbf{X}_{2:K}^T\right) + \frac{1}{2} \mathbf{x}_1^T \mathbf{x}_1 \\ & + \frac{1}{2} \text{trace}\left(\mathbf{K}_q^{-1} \mathbf{Q} \mathbf{W}^2 \mathbf{Q}^T\right) \\ & + \sum_j \ln(\beta_j) + \frac{1}{2\kappa^2} \text{trace}(\mathbf{W}^2) + \sum_j \ln \alpha_j. \end{aligned} \quad (22)$$

There are many proposed methods for solving this equation: maximum a posteriori (MAP) estimation, balanced GPDM methods, or two-stage MAP. We use balanced GPDM [48] where the hyper-parameters are initialized and held fixed, while latent positions are optimized. Hyper-parameters are then refined, while

holding latent positions fixed. This process iterates until an error threshold is reached or a maximum number of iterations occurs.

After optimizing the hyper-parameters and latent states, we can propagate the latent space dynamics using the distribution,

$$\mathbf{x}_k \sim \mathcal{N}\left(\mu_X(\mathbf{x}_{k-1}); \sigma_X^2(\mathbf{x}_{k-1}) \mathbf{I}\right), \quad (23)$$

where μ_X is the mean function and σ_X^2 is the covariance function which are defined as

$$\mu_X(\mathbf{x}) = \mathbf{X}_{2:K}^T \mathbf{K}_x^{-1} \mathbf{k}_X(\mathbf{x}), \quad (24)$$

$$\sigma_X^2(\mathbf{x}) = \phi(\mathbf{x}, \mathbf{x}) - \mathbf{k}_X(\mathbf{x})^T \mathbf{K}_x^{-1} \mathbf{k}_X(\mathbf{x}). \quad (25)$$

where $\mathbf{k}_X(\mathbf{x}) = [\phi(\mathbf{x}_2, \mathbf{x}), \dots, \phi(\mathbf{x}_K, \mathbf{x})]$. Once trajectories have been simulated in latent space, they can be projected into state space to generate simulated motions.

To project latent trajectories back into state space the mean and covariance functions can be written as

$$\mu_Q(\mathbf{x}) = \mathbf{Q}^T \mathbf{K}_q^{-1} \mathbf{k}_Q(\mathbf{x}), \quad (26)$$

$$\sigma_Q^2(\mathbf{x}) = \psi(\mathbf{x}, \mathbf{x}) - \mathbf{k}_Q(\mathbf{x})^T \mathbf{K}_q^{-1} \mathbf{k}_Q(\mathbf{x}), \quad (27)$$

where $\mathbf{k}_Q(\mathbf{x}) = [\psi(\mathbf{x}_1, \mathbf{x}), \dots, \psi(\mathbf{x}_K, \mathbf{x})]$. Note that the projected trajectories will be zero mean due to the mean subtraction applied to the original data.

5 Results

5.1 Synthetic data

Before testing our algorithm on experimental data, numerical experiments were conducted with synthetic data to compare the accuracy of the CI, ISC, and standard UKF estimators. Synthetic data was created using the same kinematics defined in Sect. 5. Joint angles were assumed to be sinusoidal with different phase shifts and amplitudes for each degree of freedom. Synthetic measurements were created in 21 cameras assuming no noise and no occlusions within the FOV of each camera. All cameras were given nonlinear lens distortion on the order of that identified by the MultiCamSelfCal toolbox for the experiments. Figures 3, 4, and 5 show the estimation error for each group of states for the synthetic data. In the figures, solid lines represent the error in our ISC UKF, dashed lines are

the error in the CI UKF, and dotted lines indicate the error in the standard UKF.

In Fig. 3, both of the spatially recursive estimators perform equally well, while the standard estimator has the largest error. Both spatially recursive estimators perform equally well because the body points used to estimate these states move almost linearly within the images: the only nonlinearity is due to the lens distortion which is removed prior to estimation. In Fig. 4, group 2 estimation errors are shown. Again, the standard estimator performs poorly for these states. The CI estimate also has a slightly larger error than the ISC

estimate for portions of the flap cycle which are not locally linear.

While there is minimal difference in the state estimates for the first two groups, states in group five show significant differences in the CI and ISC estimators, while the standard estimator still performs poorly. Figure 5 shows the estimation error for four phalangeal degrees of freedom which are in the fifth group of links. While all three estimators have periodically varying error, the ISC estimator has consistently lower error than the standard or CI UKF algorithms. The periodic nature of the estimation error is expected due to the sinusoidal form of ground truth states; the estimators

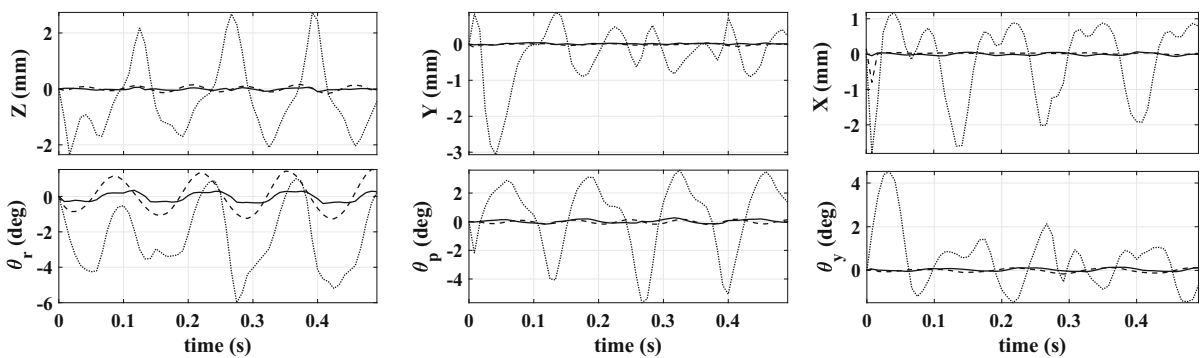


Fig. 3 Estimation error of Group 1. Both the CI (dashed line) and the ISC (solid line) estimators perform equally well for these states, while the standard estimator (dotted line) deviates markedly from the ground truth state

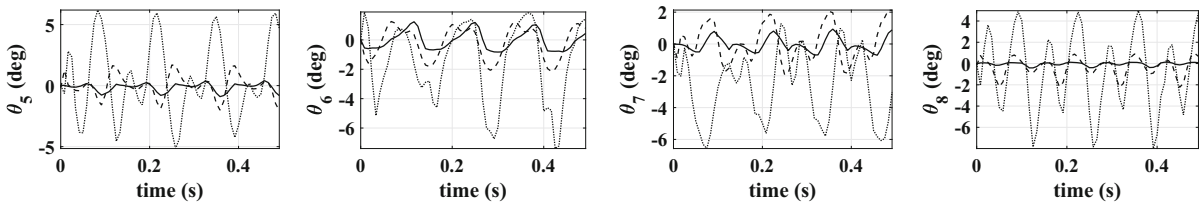


Fig. 4 Estimation error of Group 2. The ISC estimator (solid line) starts to outperform the CI estimator (dashed line). The standard estimator (dotted line) still exhibits large errors in the state estimates

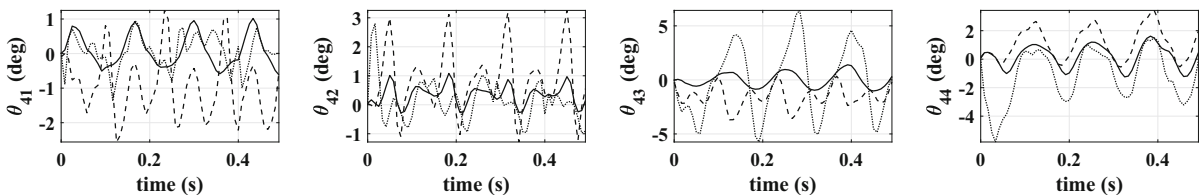


Fig. 5 Estimation error of Group 5. The ISC estimator (solid line) is clearly outperforming the CI (dashed line) and standard (dotted line) estimators. The image space motion of the features

on these links is highly nonlinear due to the kinematics of the bat skeleton

have the highest error for the segments which are not locally linear. Additionally, the measurements used to update these states in the fifth group are observations of the wing tip points for each digit. These points undergo highly nonlinear, large-baseline motions within the images which is difficult to approximate with the weak motion prior and the linear update of the Kalman filter. Additionally, any estimation error from the previous groups will be contained within the estimates for links outboard along the kinematic chain. The ISC estimator produces substantially better state estimates than the CI or standard UKF estimates because it corrects inboard link state estimates as it estimates outboard link states to insure that cross-covariances between link states are incorporated in the final state estimate. Thus, the ISC estimator provides the best estimates for this specific class of problem.

While the ISC estimator does improve estimation accuracy, it is notably slower than the CI or standard UKF estimators. In repeated trials, the ISC estimator requires twice as much computation time as the standard estimator and ten times as much computation time as the CI estimator. Because of the added computational cost of our method, we wish to insure that this strategy provides a significant improvement in state prediction accuracy for our problem. While we cannot collect experimental data at higher frame rates with our hardware, we can generate synthetic data at any frame rate we choose.

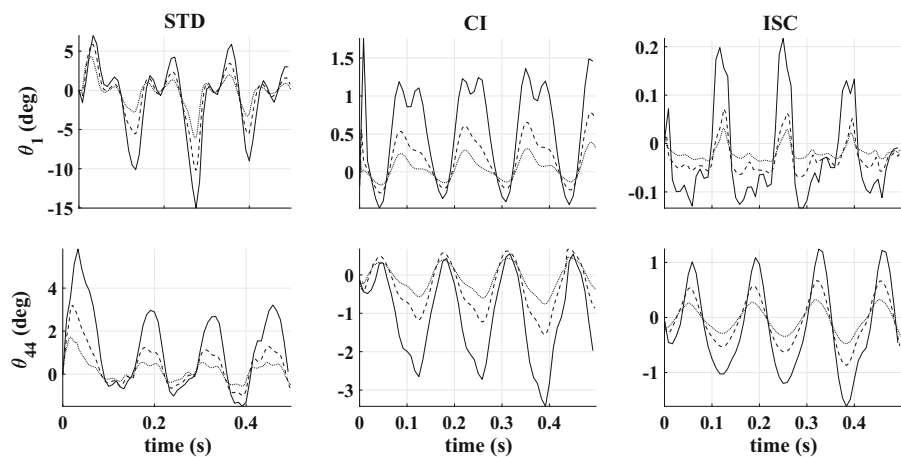
Figure 6 shows the estimation error for all three algorithms as frame rate is increased. The figure shows the error of the standard estimator in the left column, the CI estimator in the middle column, and our ISC estimator

in the right most column. The states shown are: θ_1 on the top row and θ_{44} on the bottom row. Solid lines represent 120 fps, dashed lines represent 240 fps, and dotted lines represent 480 fps. At a frame rate of 120 fps, the ISC implementation of the UKF produces state estimates which have an order of magnitude lower error than the standard or CI estimates. However, all three estimators have approximately the same error at a sampling rate of 480 fps. The weak motion model and the linear correction step of the Kalman filter induce less error during estimation as frame rate increases. As shown in the figure above, the ISC UKF does not provide a significant improvement on the estimation accuracy at high sampling rates. However, due to the improved estimation accuracy of the ISC estimator at low frame rates, we can conclude that it is well suited for our specific class of problem—capturing fast articulated motion with low-frame-rate cameras.

5.2 Experimental estimation results

The spatially recursive UKF derived above is applied to bat flight motion capture data collected at the Shandong University-VT International Laboratory at Shandong University in Jinan, China, in the summer of 2016. The experiment was conducted using 21 GoPro Hero 3+ cameras recording 720p \times 1280p video at 120 fps. The bat specimen used in the testing is a male *Hipposideros pratti*. This paper presents trajectory estimation, 3D reconstruction of flight motion, and reprojection error results for 34 fiducial markers located on the 25 rigid bodies in our kinematic definition of the bat skeleton.

Fig. 6 Effect of sample rate on estimated states for synthetic data. The sample rates are: 120 fps (solid), 240 fps (dashed), and 480 fps (dotted). Top row shows θ_1 , and bottom row shows θ_{44} . From left to right: standard, CI, and ISC error estimates



All link names, DH parameters, and estimation group assignments are included in a previous paper [5].

Before the motion capture experiments were conducted, the system was calibrated using the MultiCam-SelfCal toolbox [37]. Calibration videos were taken at the aforementioned resolution and frame rate. During this video, all lights are turned off and a single point of light was moved through the capture volume. The toolbox finds the light in the images and performs a sparse bundle adjustment algorithm to determine the intrinsic and extrinsic parameters of the cameras. The distortion parameters were used to remove radial distortion from feature points prior to state estimation. Seventeen of the twenty-one cameras in the experiment are used to produce the reconstruction results: the omitted cameras have reprojection errors higher than 2 pixels as computed by the MultiCamSelfCal toolbox.

After the calibration, feature correspondence is computed. The authors have attempted tracking points in image space using standard methods such as SIFT, SURF, and bootstrapped optical flow without much success. The low sampling rate of our cameras leads to large displacements between frames. Thus, the non-rigid motion of features is substantial and the assumptions made in these tracking methods are not valid at least without further modification or tuning. Thus, the images are labeled by hand to ensure that correct correspondence is achieved. Once the correspondences are available, trajectory estimation is performed. Figures 7, 8, and 9 show the state estimates for the three estimators on each of the state groups 1, 2, and 5. Groups 3 and 4 are omitted for brevity. Solid lines represent the ISC estimator, dotted lines represent the CI estimator, and

dashed lines represent the standard UKF. Note that in the first group of states which are shown in Fig. 7, the CI and ISC estimators return comparable estimates, while the standard estimator contains significantly more noise and diverges from the other two estimates.

In Fig. 8, the CI and ISC estimators perform comparably well for all states. The standard estimator deviates substantially from the other two estimates. Figure 9 shows the ISC estimator performing substantially better than the CI or standard estimators for all joint states in group 5. Additionally, the ISC estimator does not contain large DC offsets in joint angle which are present in the other two estimators.

Ultimately, the results above are only a qualitative analysis of the estimation accuracy, because we do not have ground truth data for our experiments. To quantitatively evaluate the estimation accuracy we reproject the identified skeleton pose into the images and determine its deviation from the original image features. Figure 10 shows the skeleton estimate from each of the three estimators reprojected into the image at 5 different time steps. Both recursive estimators perform much better than the standard estimator. The ISC estimator produces reprojections which are closer to the original feature space measurement for points in the hands of the bat. These reprojected skeletons are only a snapshot of the reprojection error.

To illustrate the estimation accuracy more completely, we compute the reprojection errors of all points at all time steps. The reprojection errors in three different cameras are shown in Fig. 11. In the figure, ISC UKF reprojection errors are denoted by the ‘+’ markers and CI UKF errors are denoted by the ‘o’ markers

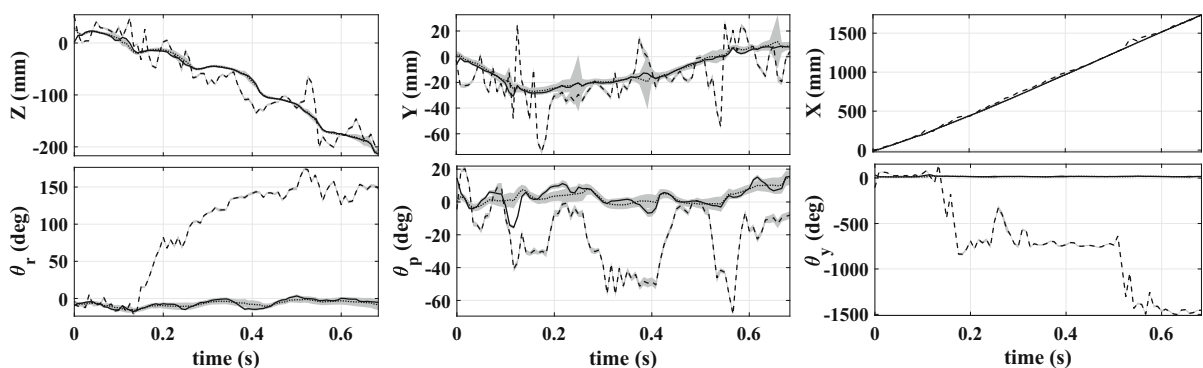


Fig. 7 Group 1 state estimates. Top row: Z , Y , and X body displacement. Bottom row: roll, pitch, and yaw angles. The ISC (solid) and CI (dotted) estimates are almost equivalent. Large

amounts of high-frequency noise are present in the standard estimator (dashed). Gray-shaded regions represent $\pm 2\sigma$ deviation

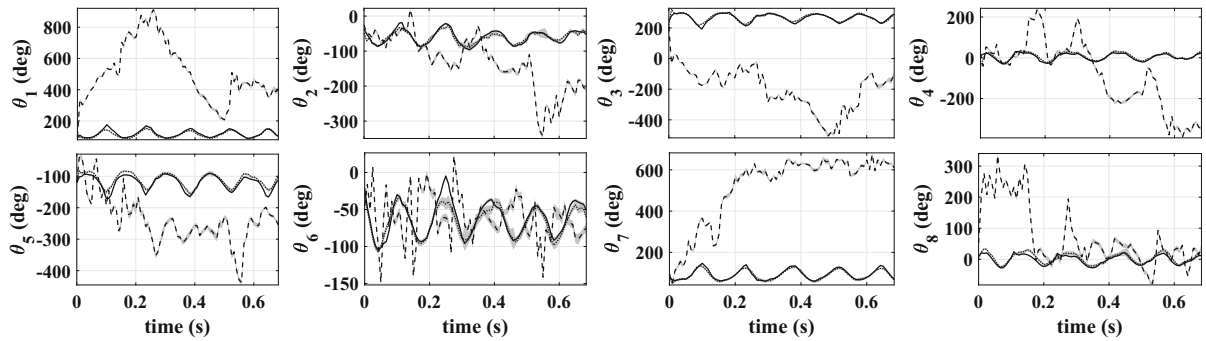


Fig. 8 Group 2 state estimates. $\theta_1 - \theta_4$: right arm joint angles. $\theta_5 - \theta_8$: left arm joint angles. ISC method (solid), standard (dashed), and CI (dotted) estimates. Gray-shaded regions represent $\pm 2\sigma$ deviation

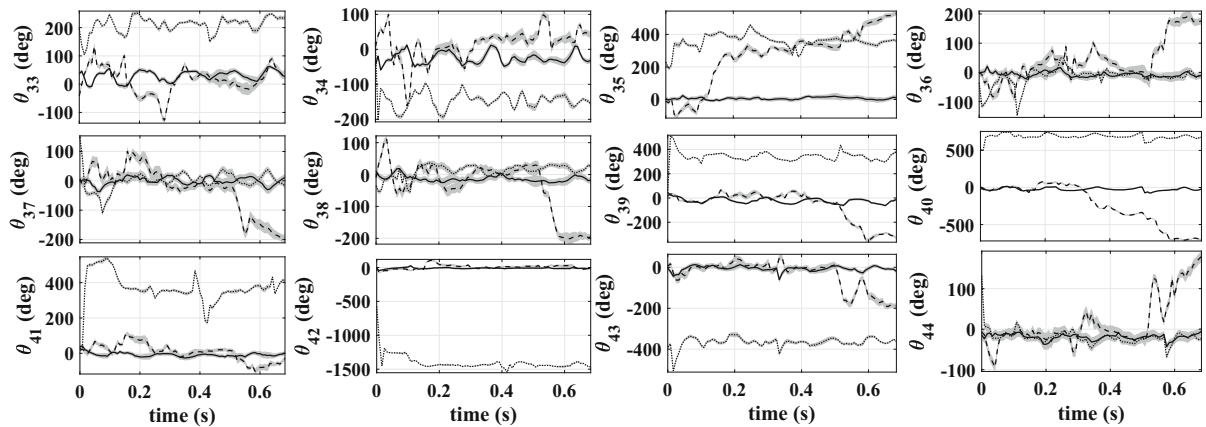


Fig. 9 Group 5 state estimates. $\theta_{33} - \theta_{38}$: left-hand second phalanx joint angles. $\theta_{39} - \theta_{44}$: right-hand second phalanx joint angles. ISC (solid), standard (dashed), and CI (dotted) estimates. Gray-shaded regions represent $\pm 2\sigma$ deviation

ers. Additionally, 95% confidence ellipses for the ISC method (solid line) and CI (dashed line) are shown. The standard estimator has been omitted from these plots because the reprojection errors are typically greater than ± 50 pixels. The uncertainty ellipses for the ISC estimator are substantially smaller than the ellipses for the CI estimator. In the three cameras presented, the reprojection error for the ISC estimator is approximately ± 5 pixels and the reprojection error for the CI estimator is approximately ± 15 pixels. Thus, the ISC estimator is performing much better than the CI estimator.

5.3 Gaussian process dynamic model results

With reliable estimates for joint states, we learn a GPDM for prediction in future tracking applications.

To perform the manifold estimation, we omit the base body translational DOFs and run the GPDM algorithm on the remaining 48 generalized joint coordinates. This generates a model which is independent of the specific path of the bat. For this paper we were able to supply the algorithm with 80 time steps (5 flap cycles) of data. To perform optimization of Eq. 22 we choose the dimensionality of the latent space, D , and initialize the latent variables, \mathbf{X} , by using the first D modes of a PCA decomposition of the original data, \mathbf{Q} .

The identified manifolds are shown in Fig. 12. In the figure, all manifolds use the RBF plus linear kernel from Eq. 18 for the dynamics and the RBF kernel in Eq. 15 for the mapping between latent space and feature space. Subfigures (a), (b), and (c) assume 2, 3, and 4 latent DOFs, respectively. The blue points are the identified latent trajectories, red points are the simulated trajectories, and green points are HMC samples on the

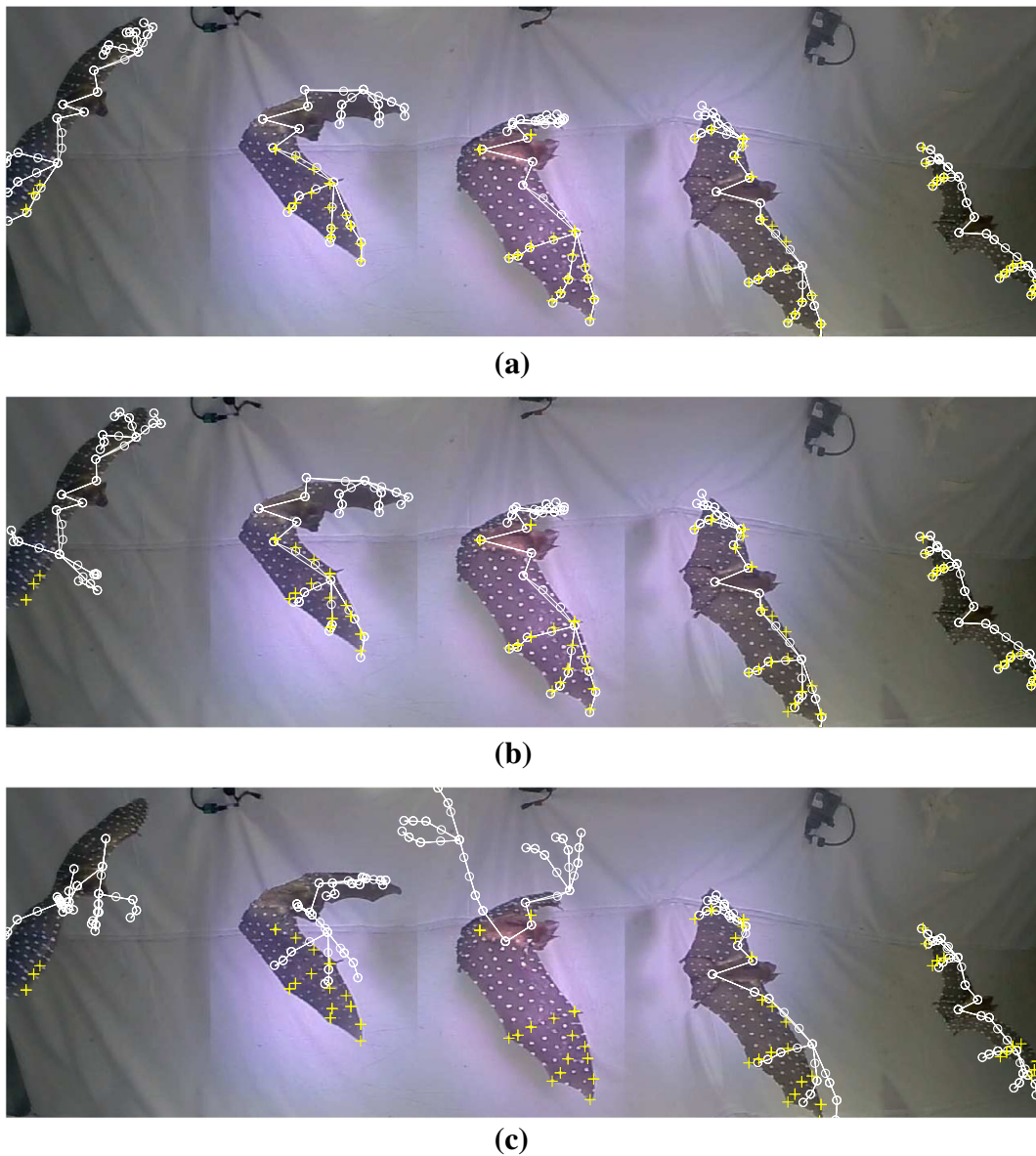


Fig. 10 Skeleton reprojection camera 2. The skeleton is reprojected back into the image at 5 different time steps. The ISC estimator produces more accurate skeletal reprojections than the other two estimators

simulated trajectories to represent the uncertainty in the manifold. Arrows indicate the direction of motion. Ideally, the manifold should appear to be cyclic and have little variation between cycles. Figure 12a shows the 2 DOF manifold. It displays large jumps in the trajectory that are not consistent from cycle to cycle. Figure 12b shows a smooth manifold with little uncertainty which is demonstrated by the close proximity of the HMC sample trajectories to the simulated trajec-

tory. However, the motion is only semi-periodic. Further discussion of the semi-periodic behavior is provided below. Finally, Fig. 12c shows the 4 DOF latent space. This manifold shows substantial uncertainty in the third DOF, and the motion in the fourth DOF is not cyclic. Due to the semi-periodic nature of the 3 DOF manifold and the consistency of the HMC samples, we assume that this is the correct number of DOF for the latent space of the observed motion.

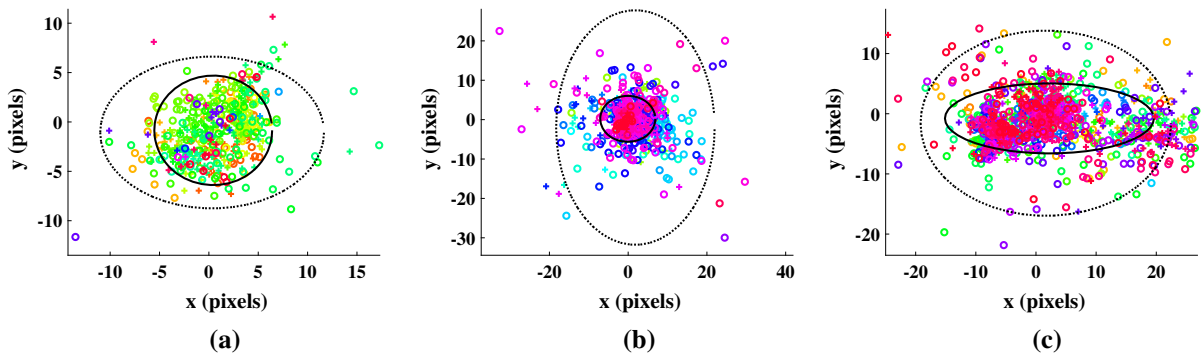


Fig. 11 Reprojection error from cameras 1–3. The reprojection errors for the ISC estimator (+) with 95% confidence ellipse (solid line) are notably smaller than the reprojection errors for the

CI estimator (o) with 95% confidence ellipse (dotted line). The standard UKF estimator is omitted because reprojection errors are typically greater than ± 50 pixels

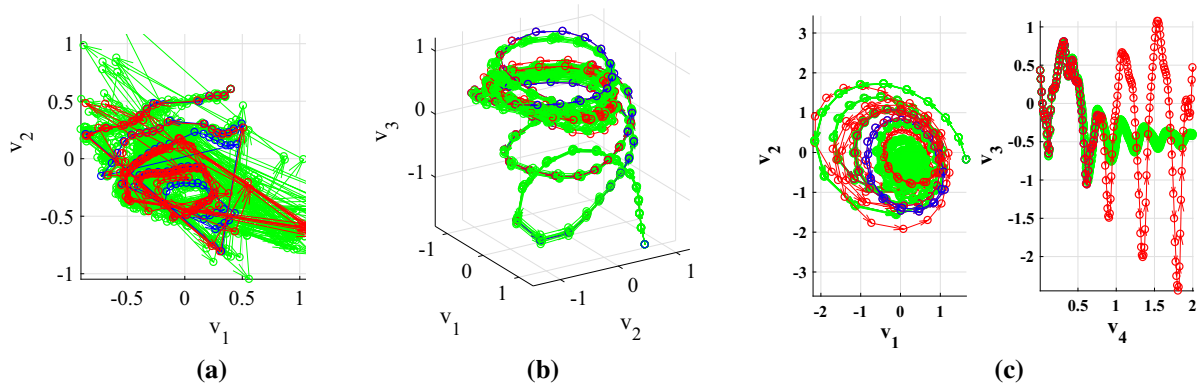


Fig. 12 Latent space visualization. Blue points are the learned latent coordinates, red points are simulated trajectories, and green points are fair HMC samples from the distribution. Arrows indicate the direction of motion. (Color figure online)

To provide an intuitive interpretation of the identified 3 DOF manifold, Fig. 13 shows the identified latent trajectory (black line) with 6 highlighted latent states that approximately encompass the quasi-circular shape of the manifold when projected onto the V_1 – V_2 plane. The latent state highlighted by the red circle corresponds to the bottom of the upstroke. The red skeleton in the middle of Fig. 13 depicts the pose of the bat during this instant of the flap cycle. Proceeding counter-clockwise around the circle, the green circle denotes the middle of the upstroke. Continuing on to the cyan circle, the bat reaches the top of the upstroke and the downstroke begins at the yellow circle. Finally, the magenta circle corresponds to the middle of the downstroke and the blue circle to the bottom of the downstroke. From the color-corresponded skeletal projections in this figure, we can conclude that the V_1 and

V_2 directions describe the cyclic nature of bat flapping flight and these latent states describe what portion of the flap cycle the bat is in. To further understand the identified dynamic model, Fig. 14 shows the identified manifold in the center and four highlighted points which move up the V_3 direction. The circular points correspond to the bottom of the upstroke, and the triangles correspond to the top of the upstroke. As the colors get lighter the adduction–abduction degree of freedom, θ_2 , becomes less extreme. Thus, for lower values of V_3 the amplitude of flapping is smaller than for larger values. In the experiment conducted for this paper, the lower values of V_3 occur at the beginning of the test and the higher values at the end of the test. We can further understand the implications of this phenomenon by projecting these latent states back into state space.

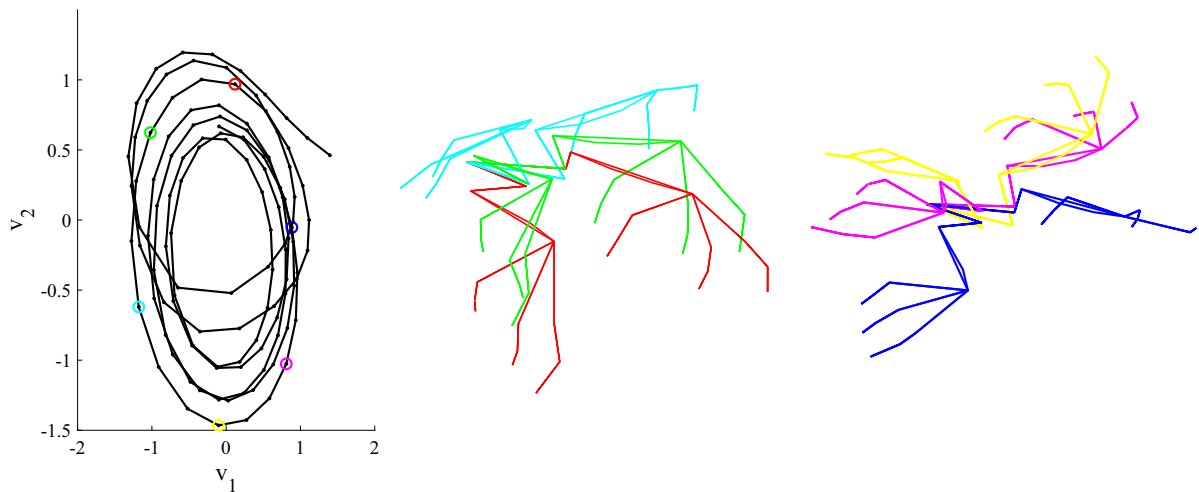


Fig. 13 Understanding the latent space: V_1 and V_2 . The identified manifold is depicted at the left. The circle marks are color corresponded to the skeletons in the center and left figures. Starting at the red circle, the wings are approximately at the bottom of the upstroke. The upstroke continues counter-clockwise around

the quasi-circular manifold. The top of the upstroke is indicated by the yellow marker, and the bottom of the downstroke is indicated by the blue marker. Thus, the V_1 and V_2 directions describe the portion of the flap cycle the bat is in. (Color figure online)

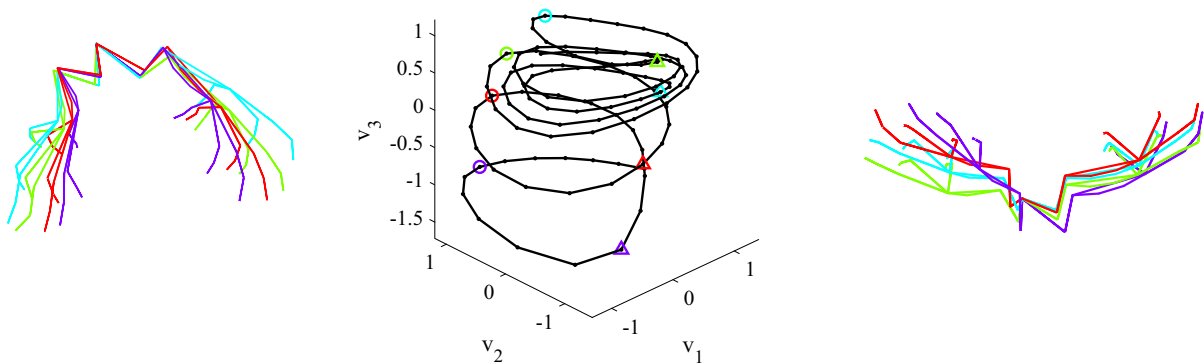


Fig. 14 Understanding the latent space: V_3 . In the center figure, circle markers are placed at latent states which correspond to the bottom of the downstroke and triangle markers are placed at locations which correspond to the top of the downstroke. As the markers get lighter in color, V_3 increases. These markers are

color corresponded to the skeletal plots on the left and right. Skeletons that are lighter show smaller flapping amplitudes in the adduction–abduction degree of freedom. Thus, lower values of V_3 correspond to the bat flapping ‘harder’ than higher values of V_3 . (Color figure online)

The projection of latent states into state space is shown in Fig. 15. The predictions (solid and dashed lines) match the experimental data (dots) very closely which further demonstrates the consistency of the identified 3 DOF manifold. Note that the solid line cycles for $\theta_1 - \theta_3$ at the beginning of the test have a larger amplitude than the remaining dashed line cycles. This

behavior is consistent with the previous discussion of the V_3 dimension of the identified manifold and is likely the cause of the semi-periodic latent space trajectories discussed before. Additionally, the extrapolated data (dotted line) appears to be periodic repetitions of the experimental data.

Fig. 15 Feature space projection of latent trajectories. θ_1 through θ_4 is shoulder and elbow rotations for the right wing. θ_{41} – θ_{44} are second phalanx rotations on the right hand. Dots represent the training data points. The first two flap cycles (solid lines) are of notably larger magnitude than the remaining cycles of training data (dashed lines). The remaining cycles (dotted lines) are extrapolated from the experimental data using the learned GPDM

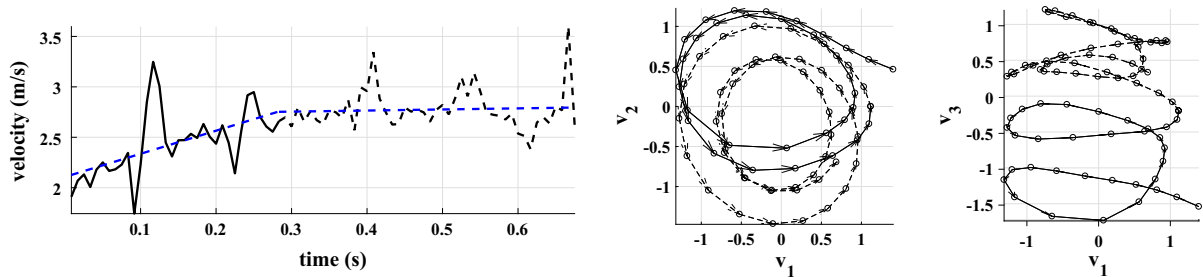
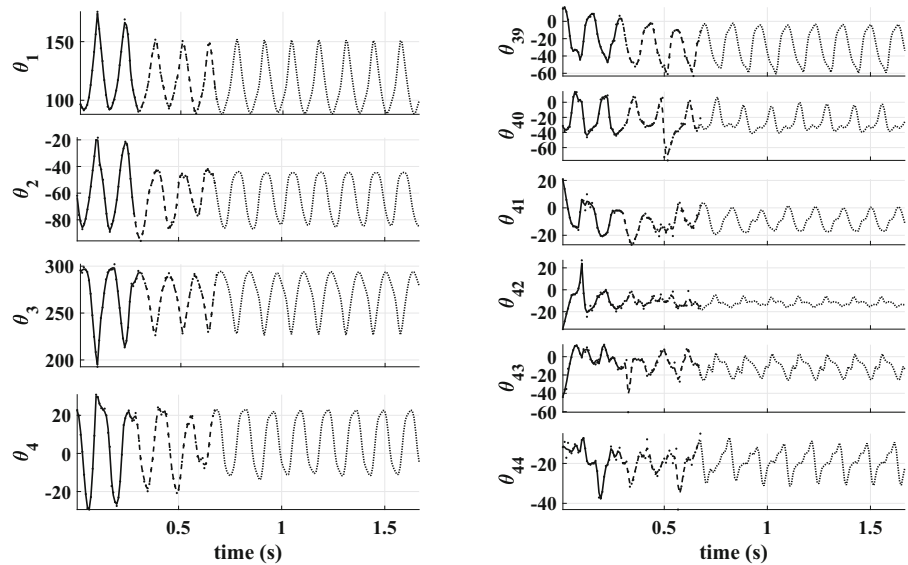


Fig. 16 [Left] magnitude of body velocity. [Middle and right] two views of the manifold. The bat is clearly accelerating (solid lines) for the first half of the test and coasting (dashed lines) for

the remainder. These line types correspond to the line types used in Fig. 15

Although the latent coordinates project correctly into state space, it is still important to determine the significance of the aperiodic nature of motion in the V_3 dimension of the latent space. If the bat performed each flap cycle exactly the same, there should be no deviation between cycles in state space or in the latent space. Because of the semi-periodic nature of the data, we posit that the bat is not performing steady, straight, and level flight. By computing the linear velocity of the bat's body, which is shown in Fig. 16, we can clearly see that the bat is accelerating for the first 0.3 s of the experiment and is coasting for the remainder. The large amount of noise in the velocity profile is induced by the first-order derivative used to calculate velocity. The left of Fig. 16 shows the body velocity which is line type corresponded to the two manifold views on the bottom. In the solid line section, the bat is accel-

erating. In the dashed line section, the bat is coasting. There are two clearly different segments to the manifold which is consistent with the estimated states and the body velocity. Thus, we have successfully identified a discrete dynamic model for straight and level bat flight.

6 Conclusions

The first goal of this paper is to estimate the inertial trajectories of fiducial markers on a bat wing and identify the generalized joint coordinates during straight and level flapping flight of a bat. This paper introduces a methodology which exploits the tree topology of the bat skeletal kinematics to formulate a conditionally independent Bayes' filter which incorporates inboard state correction (ISC) for trajectory estimation. We present

our definition of the bat skeletal kinematics, a recursive Bayes' filter with inboard state correction for estimating joint coordinates, and the sensor and motion models used in the implementation. Our method is shown to yield better state estimates on noise-free synthetic data and on experimental data than the standard or conditionally independent without inboard state correction estimators. Furthermore, inertial trajectories of fiducial markers were reconstructed with ± 5 pixel reprojection error using the ISC estimator.

The second goal of this work is to derive a low-dimensional dynamic model suitable for analysis and study of flapping flight. We employ Gaussian process dynamic models (GPDMs) to learn both a latent space representation for bat flight dynamics and a mapping from latent space back to joint space. This is—to the authors' knowledge—the first nonlinear dimensionality reduction of bat flapping flight. We have successfully identified a model, corresponding to an underlying low-dimensional manifold, which closely resembles the experimental data provided and produces plausible synthesized motions.

In the future, this model will be used in conjunction with an estimation strategy that respects the geometry of the identified manifold. This will improve the accuracy of predicted marker locations substantially and should help improve the accuracy with which feature correspondences can be established. Furthermore, it is very difficult to induce a bat to perform exactly the same maneuver more than once. The authors would like to determine a way to identify small deviations from a typical flap cycle so that a periodic manifold can be identified from semi-periodic data.

Acknowledgements The authors would like to thank the following project teams from the Shandong University Virginia Tech (SDU-VT) International Laboratory's summer 2015 and summer 2016 research collaboration: Bat Flight Motion Capture Senior Design Project, SDU-VT Flight Team, and the SDU-VT Bat Husbandry Team. The authors would like to give a special thanks to the following individuals for their extra diligence in completing this project: Yang Xu, Zhao Yanan, Wang Chenhao, Sha Qiyuan, Wang Bingcheng, Lin Youxi, Ma Zhiqiang, Zhang Liujun, Cao Ze, Wang Jinzhen, Laura Bunn, Eric Anderson, Kenton Anderson, and Hunter McClelland. Additionally we would like to thank Sharon Swartz, Andrea Rummel, and Lawrence Wang from Brown University for supporting experiments and loan of critical equipment. We would also like to thank Orangkucing Labs for developing the MewPro open-source GoPro camera control system (<http://mewpro.cc/>). Finally, we would like to thank the National Natural Science Foundation of China (Grant Nos. 11374192 and 11574183); Fundamental Research Fund of

Shandong University (Grant No. 2014QY008); Minister of Education of China Tese grant for faculty exchange; US National Science Foundation (Grant No. 1510797); and Virginia Tech Institute for Critical Technology and Applied Science (ICTAS, through support for the BIST Center).

References

1. Aldridge, H.D.: Kinematics and aerodynamics of the greater horseshoe bat, *Rhinolophus ferrumequinum*, in horizontal flight at various flight speeds. *J. Exp. Biol.* **126**, 479–97 (1986)
2. Alvarez, M.A., Rosasco, L., Lawrence, N.D.: *Kernels for Vector-valued Functions: A Review*. Now Publishers Inc., Dordrecht (2012)
3. Bahlman, J.W., Price-Waldman, R.M., Lippe, H.W., Breuer, K.S., Swartz, S.M.: Simplifying a wing: diversity and functional consequences of digital joint reduction in bat wings. *J. Anat.* **229**(1), 14–27 (2016)
4. Bender, M.J., McClelland, H.G., Bledt, G., Kurdila, A.J., Furukawa, T., Mueller, R.: Trajectory estimation of bat flight using a multi-view camera system. In: *AIAA SciTech*, AIAA, Kissimmee, FL, USA, pp. 1–13 (2015)
5. Bender, M.J., McClelland, H.G., Kurdila, A.J., Müller, R.: Recursive Bayesian estimation of bat flapping flight using kinematic trees. In: *AIAA SciTech*, AIAA, San Diego, CA, USA, pp. 1–12 (2016)
6. Bender, M.J., McClelland, H.M., Kurdila, A., Müller, R.: Recursive Bayesian estimation of bat flapping flight using kinematic trees. In: *AIAA Modeling and Simulation Technologies Conference* (January), pp. 1–12 (2016)
7. Bergou, A.J., Swartz, S., Breuer, K., Taubin, G.: 3D reconstruction of bat flight kinematics from sparse multiple views. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1618–1625 (2011)
8. Bradski, G.: *OpenCV Library*. Dr. Dobbs' Journal of Software Tools (2000)
9. Canton-Ferrer, C., Casas, J.R., Pardas, M.: Towards a low cost multi-camera marker based human motion capture system. In: *2009 16th IEEE International Conference on Image Processing (ICIP)*, pp. 2581–2584 (2009)
10. Cheney, J.A., Ton, D., Konow, N., Riskin, D.K., Breuer, K.S., Swartz, S.M.: Hindlimb motion during steady flight of the lesser dog-faced fruit bat. *Cynopterus brachyotis*. *PLoS ONE* **9**(5), 1–8 (2014)
11. Colorado, J., Barrientos, A., Rossi, C., Breuer, K.S.: Biomechanics of smart wings in a bat robot: morphing wings using SMA actuators. *Bioinspiration Biomim.* **7**(3), 36,006 (2012)
12. Dellaert, F., Seitz, S., Thrun, S., Thorpe, C.: Feature correspondence : a Markov chain Monte Carlo approach. In: *Advances in Neural Information Processing Systems*, pp. 852–858 (2001)
13. Erol, A., Bebis, G., Nicolescu, M., Boyle, R.D., Twombly, X.: A review on vision-based full DOF hand motion estimation. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, p. 75 (2005)
14. Gopalakrishnan, P., Tafti, D.K.: Effect of wing flexibility on lift and thrust production in flapping flight. *AIAA J.* **48**(5), 865–877 (2010)

15. Hauberg, S., Lauze, F., Pedersen, K.S.: Unscented Kalman filtering on Riemannian manifolds. *J. Math. Imaging Vis.* **46**(1), 103–120 (2013)
16. Hedrick, T.L.: Software techniques for two- and three-dimensional kinematic measurements of biological and biomimetic systems. *Bioinspiration Biomim.* **3**(3), 1–6 (2008)
17. Hofmann, M., Gavrilu, D.M.: Multi-view 3D human pose estimation in complex environment. *Int. J. Comput. Vis.* **96**(1), 103–124 (2012)
18. Hubel, T.Y., Hristov, N.I., Swartz, S.M., Breuer, K.S.: Time-resolved wake structure and kinematics of bat flight. *Exp. Fluids* **46**(5), 933–943 (2010)
19. Isard, M., Blake, A.: ICondensation: unifying low-level and high-level tracking in a stochastic framework. In: 5th European Conference on Computer Vision (1998)
20. Lawrence, N.: Gaussian process latent variable models for visualisation of high dimensional data. *Computer* **16**(5), 329–336 (2004)
21. Liu, Z., Zhu, J., Bu, J., Chen, C.: A survey of human pose estimation: the body parts parsing based methods. *J. Vis. Commun. Image Represent.* **32**, 10–19 (2015)
22. Ma, Y., Soatto, S., Kosecká, J., Sastry, S.S.: An Invitation to 3-D Vision: From Images to Geometric Models, 1st edn. Springer, Berlin (2004)
23. MacCormick, J., Isard, M.: Partitioned sampling, articulated objects, and interface-quality hand tracking. *Comput. Vis. ECCV* **2000**, 3–19 (2000)
24. Mackay, D.J.C.: Information Theory, Inference, and Learning Algorithms, 7.2nd edn. Cambridge University Press, Cambridge (2003)
25. Modeslund, T., Hilton, A., Kruger, V.: A survey of advances in vision-based human motion capture and analysis. *Comput. Vis. Image Underst.* **104**, 90–126 (2006)
26. Ramezani, A., Chung, S.J., Hutchinson, S.: A biomimetic robotic platform to study flight specializations of bats. *Sci. Robot.* **2**(3), 1–12 (2017)
27. Ramezani, A., Shi, X., Chung, S.J., Hutchinson, S.: Bat bot (b2), a biologically inspired flying machine. In: 2016 IEEE International Conference on Robotics and Automation (ICRA), pp. 3219–3226 (2016)
28. Riskin, D.K., Bergou, B., Breuer, K.S., Swartz, S.M.: Upstroke wing flexion and the inertial cost of bat flight. *Proc. R. Soc. B Biol. Sci.* **279**(1740), 2945–2950 (2012)
29. Riskin, D.K., Iriarte-Díaz, J., Middleton, K.M., Breuer, K.S., Swartz, S.M.: The effect of body size on the wing movements of pteropodid bats, with insights into thrust and lift production. *J. Exp. Biol.* **213**, 4110–4122 (2010)
30. Riskin, D.K., Willis, D.J., Iriarte-Díaz, J., Hedrick, T.L., Kostandov, M., Chen, J., Laidlaw, D.H., Breuer, K.S., Swartz, S.M.: Quantifying the complexity of bat wing kinematics. *J. Theor. Biol.* **254**(3), 604–615 (2008)
31. Send, W., Fischer, M., Jebens, K., Mugrauer, R., Nagarathinam, A., Scharstein, F.: Artificial hinged-wing bird with active torsion and partially linear kinematics. In: 28th Congress of the International Council of the Aeronautical Sciences, pp. 23–28 (2012)
32. Shyy, W., Aono, H., Chimakurthi, S., Trizila, P., Kang, C.K., Cessnik, C., Liu, H.: Recent progress in flapping wing aerodynamics and aeroelasticity. *Prog. Aerosp. Sci.* **46**(7), 284–327 (2010)
33. Sigal, L., Isard, M., Haussecker, H., Black, M.J.: Loose-limbed people: estimating 3D human pose and motion using non-parametric belief propagation. *Int. J. Comput. Vis.* **98**(1), 15–48 (2012)
34. Spong, M.W., Hutchinson, S., Vidyasagar, M.: Robot Modeling and Control, 1st edn. Wiley, Hoboken (2006)
35. Stenger, B., Thayananthan, A., Torr, P.H.S., Cipolla, R.: Filtering using a tree-based estimator. In: ICCV '03 Proceedings of the Ninth IEEE International Conference on Computer Vision, vol. 2, pp. 1063–1071 (2003)
36. Stenger, B., Thayananthan, A., Torr, P.H.S., Cipolla, R.: Model-based hand tracking using a hierarchical bayesian filter. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(9), 1372–1384 (2006)
37. Svoboda, T., Martinec, D., Pajdla, T.: A convenient multi-camera self-calibration for virtual environments. *Presence* **14**(4), 407–422 (2005)
38. Szeliski, R.: Computer Vision: Algorithms and Applications. Springer, Berlin (2004)
39. Thayananthan, A., Stenger, B., Torr, P., Cipolla, R.: Learning a kinematic prior for tree-based filtering. In: Proceedings of the British Machine Vision Conference, pp. 60.1–60.10 (2003)
40. The Mathworks Inc.: Computer Vision System Toolbox (2014)
41. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics. MIT Press, Cambridge (2005)
42. Tian, X., Iriarte-Díaz, J., Middleton, K., Galvao, R., Israeli, E., Roemer, A., Sullivan, A., Song, A., Swartz, S., Breuer, K.: Direct measurements of the kinematics and dynamics of bat flight. *Bioinspiration Biomim.* **1**(4), S10–S18 (2006)
43. Urtasun, R., Fleet, J.D., Fua, P.: Gaussian Process Dynamical Models for 3D people tracking. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (June), vol. 1 (2006)
44. Viswanath, K., Nagendra, K.: Climbing flight of a fruit bat deconstructed. In: 52nd Aerospace Sciences Meeting (January), pp. 1–24 (2014)
45. Viswanath, K., Nagendra, K., Cotter, J., Frauenthal, M., Tafti, D.K.: Straight-line climbing flight aerodynamics of a fruit bat. *Phys. Fluids* **26**(2), 604 (2014)
46. Viswanath, K., Tafti, D.K.: Effect of frontal gusts on forward flapping flight. *AIAA J.* **48**(9), 2049–2062 (2010)
47. Wang, J., Fleet, D., Hertzmann, A.: Gaussian process dynamical models. In: Advances in Neural Information Processing Systems, pp. 1441–1448 (2006)
48. Wang, J.M., Fleet, D.J., Hertzmann, A.: Gaussian Process Dynamic Models MatLab Code (2008). www.dgp.toronto.edu/~jmwang/gpdm/
49. Wang, J.M., Fleet, D.J., Member, S., Hertzmann, A.: Gaussian process dynamic models for human motion. *IEEE Trans. Pattern Anal. Mach. Intell.* **30**(2), 283–298 (2008)
50. Wolf, M., Johansson, L.C., von Busse, R., Winter, Y., Hedenström, A.: Kinematics of flight and the relationship to the vortex wake of a Pallas' long tongued bat (*Glossophaga soricina*). *J. Exp. Biol.* **213**(12), 2142–2153 (2010)
51. Wood, R.J.: The first takeoff of a biologically inspired at-scale robotic insect. *IEEE Trans. Robot.* **24**(2), 341–347 (2008)