Modeling and Estimation of Bat Flight for Learning Robotic Joint Geometry from Potential Fields

Matthew J. Bender

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Mechanical Engineering

Andrew J. Kurdila, Co-chair
Rolf Müller, Co-chair
Steve Southward
Craig Woolsey
Tomonari Furukawa

September 21, 2018
Arlington, VA

Keywords: Biomotion Studies, Motion Capture, Empirical Potential Functions, Kalman Filter, State Estimation, Guassian Process Dynamic Modeling, Bioinspired Design

Modeling and Estimation of Bat Flight for Learning Robotic Joint Geometry from Potential Fields

Matthew J. Bender

(ABSTRACT)

In recent years, the design, fabrication, and control of robotic systems inspired by biology has gained renewed attention due to the potential improvements in efficiency, maneuverability, and adaptability with which animals interact with their environments. Motion studies of biological systems such as humans, fish, insects, birds and bats are often used as a basis for robotic system design. Often, these studies are conducted by recording natural motions of the system of interest using a few high-resolution, high-speed cameras. Such equipment enables the use of standard methods for corresponding features and producing three-dimensional reconstructions of motion. These studies are then interpreted by a designer for kinematic, dynamic, and control systems design of a robotic system. This methodology generates impressive robotic systems which imitate their biological counter parts. However, the equipment used to study motion is expensive and designer interpretation of kinematics data requires substantial time and talent, can be difficult to identify correctly, and often yields kinematic inconsistencies between the robot and biology.

To remedy these issues, this dissertation leverages the use of low-cost, low-speed, low-resolution cameras for tracking bat flight and presents a methodology for automatically learning physical geometry which restricts robotic joints to a motion submanifold identified from motion capture data. To this end, we present a spatially recursive state estimator which incorporates inboard state correction for producing accurate state estimates of bat flight. Using these state estimates, we construct a Gaussian process dynamic model (GPDM) of bat flight which is the first nonlinear dimensionality reduction of flapping flight in bats. Additionally, we formulate a novel method for learning robotic joint geometry directly from the experimental observations. To do this, we leverage recent developments in learning theory which derive analytical-empirical potential energy fields for identifying an underlying motion submanifold. We use these energy fields to optimize a compliant structure around a single degree-of-freedom elbow joint and to design rigid structures around spherical joints for an entire bat wing. Validation experiments show that the learned joint geometry restricts the motion of the joints to those observed during experiment.

Modeling and Estimation of Bat Flight for Learning Robotic Joint Geometry from Potential Fields

Matthew J. Bender

(GENERAL AUDIENCE ABSTRACT)

In recent years, robots modeled after biological systems have become increasingly prevalent. Such robots are often designed based on motion capture experiments of the animal they aim to imitate. The motion studies are typically conducted using commercial motion capture systems such as Vicon™ or OptiTrack™ or a few high-speed, high-resolution cameras such as those marketed by Photron™ or Phantom™. These systems allow for automated processing of video sequences into three-dimensional reconstructions of the biological motion using standard image processing and state estimation techniques. The motion data is then used to drive robotic system designs such as the Sony™ Aibo™ dog and the Boston Dynamics Atlas humanoid robot. While the motion capture data forms a basis for these impressive robots, the progression from data to robotic system is neither algorithmic nor rigorous and requires substantial interpretation by a human.

In contrast, this dissertation presents a novel experimental and computational framework which uses low-speed, low-resolution cameras for capturing the complex motion of bats in flight and introduces a methodology which uses the motion capture data to directly design geometry which restricts the motion of joints to the motions observed in experiment. The advantage of our method is that the designer only needs to specify a general joint geometry such as a ball or pin joint, and geometry which restricts the motion is automatically identified. To do this, we learn an energy field over the set of kinematic configurations observed during experiment. This energy field "pushes" system trajectories towards those experimentally observed trajectories. We then learn compliant or rigid geometry which approximates this energy field to physically restrict the range of motion of the joint. We validate our method by fabricating joint geometry designed using both these approaches and present experiments which confirm that the reachable set of the joint is approximately the same as the set of configurations observed during experiments.

Dedication

To my parents, Ken and Susan.

Acknowledgments

Firstly, I would like to thank my advisors, Andrew Kurdila and Rolf Müller, for their excellent leadership in this long journey. You cultivated an environment in which I could explore possibilities and grow as a scientist...

I would also like to thank my research team. In particular the SDU-VT International Lab and all those who have contributed so much to the experiments and processing of data upon which this thesis is founded. I cannot begin to name all of those who helped, so please insert your name here...

Finally, I would like to thank John Nafziger. The work we did at Embry-Riddle Aeronautical University's Prescott campus forever changed the course of my career. You are a sounding board, and life long friend.

If I have learned anything through the course of my academic career it is that ...

"*The simple solution is generally the correct one.*"

-Occam's Razor

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, many investigators have used biology as inspiration for robotic systems due to the promises of increased efficiency, adaptability and maneuverability as compared to conventional robotic platforms such as wheeled, tracked, fixed-wing, or rotary-wing systems. The current design paradigm for designing biologically inspired systems consists of three main steps: 1) record observations of the motion of a biological system using commercially available motion capture systems or a few high-speed, high-frame rate cameras [29, 30, 50, 71, 72, 93], 2) perform a quantitative or qualitative analysis of the system of interest [87, 109, 114, 115], and 3) using designer interpretation of the motion data, design a system which mimics the observed motion [8, 75, 80]. This design paradigm has yielded many impressive and functional robotic systems [56, 77]. However, commercial motion capture systems made by Vicon™ or OptiTrack™ or high-quality cameras made by Photron™ or Phantom™ are expensive. Additionally, designer interpretation of kinematics data requires substantial time and talent and can differ from its biological counterpart in important ways [45]. Due to the proliferation or research in bioinspired robotics, a new, low-cost, algorithmic approach for collecting biomotion data and synthesizing robotic hardware is needed. To this end, this dissertation develops an experimental protocol which exploits an emerging class of low-speed, low-resolution cameras for studying the complex motions of bat flight, develops a low-dimensional nonlinear dynamic model of the observed motion, and develops an algorithmic framework for learning joint geometry from biological motion capture data.

Bat motion is difficult to study due to the articulated wing skeleton [52] and nonlinear, anisotropic, internally-actuated wing membrane [101]. To study this complex motion, many investigations have used a few high-frame-rate, high-resolution cameras [24, 86]. In contrast, our experimental protocol employs twenty to thirty GoPro Hero 3+ Black Cameras which reduces occlusions of feature points but introduces nonlinear lens distortion and large-baseline, nonlinear motions in image space. These issues make state estimation difficult because standard methods for image correspondence tend to fail when nonlinearities and large-baseline motion are present in the image sequences. Even with hand annotated images, the low sampling rate produced position and pose estimates of the bat skeleton which, when projected back into the images, exhibited very high reprojection errors. To reduce estimation errors, we derive a conditionally independent state estimator which incorporates inboard state correction. Through both synthetic and experimental results, this dissertation shows that our estimator produces state estimates which are an order of magnitude more accurate than those generated by Kalman filters which do not incorporate inboard state correction.

In addition to exploiting inexpensive imaging equipment for motion capture of bat flight, a goal of this project was to develop a low-order dynamic model of bat flight. Fortunately, there is a large body of work which developed Gaussian Process models for dimensionality reduction of human motion. Methods such as Gaussian Process Latent Variable Models (GPLVM) [59] and Gaussian Process Dynamic Models (GPDM) [116] have shown that human walking can be described in a two to three degree of freedom latent space. We use the GPDM framework because it identifies the latent space and the dynamics within the latent space simultaneously. Our application of GDPM to bat flight data is the first nonlinear dimensionality reduction of bat flight and provides critical insight to coupling between the degrees of freedom of the bat skeleton. This dynamic model could be used in the future as the motion model in a state estimator or for robotic system design. While dynamics and control investigations are outside the scope of this thesis, we use the kinematics data to develop rigorous methods for robotic joint design.

The current design paradigm for biologically inspired robots would require designer interpretation of the kinematics data generated by our low-cost motion capture system. In contrast, we develop two frameworks which learn range-of-motion limiting features of robotic joint geometry that restrict motions to those observed during experiment. Both methods presented herein learn a submanifold on which motions evolve by developing analytical-empirical (AE) potential energy functions over the space of admissible configurations. The AE potential functions serve to "push" system trajectories close to the set of configurations observed during experiments. Our first investigation which uses such energy models optimizes the stiffness of compliant spring elements to approximate the learned energy function. Thus the compliant structure which surrounds the joint penalizes motions away from the identified manifold by storing strain energy as the joint reaches the edge of the admissible set. Our second AE potentials paper uses the AE model to identify a zero-potential set of configurations which is used to design range-of-motion restricting end caps for spherical joints. We construct an entire bat wing using this method and show that the learned geometry successfully restricts joint motions to those observed during experiment.

This thesis is comprised of the three journal papers which describe the experimental methods for studying the complex motion of bat flight, develop a Gaussian Process Dynamic Model (GPDM) of bat flight, optimize a compliant, range-of-motion-limiting structure around a single degree of freedom joint, and learn the geometry of range-of-motion limiting end-caps from experimental observations of bat flight. The citations for the three papers and the contributions of each author are listed below

1. Matthew J Bender, Li Tian, Xiaozhou Fan, Andrew Kurdila, and Rolf Müller. Spatially recursive estimation and gaussian process dynamic models of bat flapping fight. *Nonlinear Dynamics*, (In Press), 2018

    - Matt Bender: Led development of GoPro camera array and synchronization system as well as the collection of bat flight data. Derived the conditionally independent Bayes' filter which incorporates inboard state correction. Processed

calibration data and wrote most of the MatLab code used for state estimation and processing. Developed the Gaussian Process Dynamic Model of bat flight. Generated all figures and results presented in this paper.

- Li Tian: Invaluable contributions to development of synchronization system and GoPro array. Assisted with data collection and provided editorial comments on the paper.
- Xiaozhou Fan: Invaluable contribution to data processing by leading the team which manually labeled images such that state estimation could be conducted. Assisted with data collection and provided editorial comments on the paper.
- Andrew Kurdila: Co-chair of first author's PhD Committee. Invaluable contributions to the development of the estimator and GPDM of bat flight. Provided copious editorial comments on writeup.
- Rolf Müller: Co-chair of first author's PhD Committee. Invaluable contributions to data collection both in execution and in acquiring funding and establishing facilities and equipment for experiments. Provided many editorial comments on writeup.

2. Matt J. Bender, Aishwarya George, Nathan Powell, Andrew Kurdila, and Rolf Müller. Empricial potential functions for driving bioinspired joint design. *ASME Journal of Dynamic Systems, Measurements, and Control*, In Press, 2018

    - Matt Bender: Designed the general structure of the joint geometry, derived the solution for beam displacement and force calculations, wrote a substantial portion of the paper. Fabricated and tested all prototype joints. Generated all figures and results presented in this paper.
    - Aishwarya George: Contributed to code development, wrote an FEM section for the paper which was removed from the final paper due to length restrictions, and provided editorial comments on the writeup.
    - Nathan Powell: Contributed to code development, wrote an FEM section for the paper which was removed from the final paper due to length restrictions, and provided editorial comments on the writeup.
    - Andrew Kurdila: Co-chair of first author's PhD Committee. Invaluable contributions to the initial concept that robotic hardware could be designed using analytical empirical potential energy functions. Wrote theory section which develops the AE models used in this paper. Provided copious editorial comments on writeup.
    - Rolf Müller: Co-chair of first author's PhD Committee. Invaluable contributions to data collection of bat flight both in execution and in acquiring funding and establishing facilities and equipment for experiments. Provided many editorial comments on writeup.

3. Matt J. Bender, Jia Guo, Nathan Powell, Andrew Kurdila, and Rolf Müller. Learing bioinspired joint geometry from motion caputre data of bat flight. *Bioinspiration & Biomimetics*, In Review, 2018

    - Matt Bender: Formulated the process for using the analytical empirical potential functions for learning the geometry of joint end-caps. Developed MatLab code for processing data. Designed, fabricated, and tested all hardware used in this paper. Wrote a substantial portion of the final manuscript.
    - Jia Guo: Assisted with code development and provided editorial comments on the writeup.
    - Nathan Powell: Assisted with code development and provided editorial comments on the writeup.
    - Andrew Kurdila: Co-chair of first author's PhD Committee. Invaluable contributions to the development analytical-empirical models theory and wrote the associated appendix of the paper. Provided copious editorial comments on writeup.
    - Rolf Müller: Co-chair of first author's PhD Committee. Invaluable contributions to data collection both in execution and in acquiring funding and establishing facilities and equipment for experiments. Provided many editorial comments on writeup.

Chapter 2

Spatially Recursive Estimation and Gaussian Process Dynamic Models of Bat Flapping Flight

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.

2.1 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 [120] or bird [90]. When compared to insects and birds, the potential benefits of using bats for MAV bioin-

spiration include enhanced maneuverability, stability, and efficiency. To date, researchers have synthesized 2 examples of bat-like robots. One incorporates a smart metal alloy (SMA) actuator [25] and another employs anisotropic membrane material to approximate the bat wing membrane [80, 82]. 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 Section 2.4, 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

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

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 $\boldsymbol{r}_{0,l}^0$ 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 $\boldsymbol{q}_{1:l}$ 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,

$$\mathbf{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \mathbf{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \frac{\partial V}{\partial \boldsymbol{q}} = \boldsymbol{\tau}_a + \boldsymbol{\tau}_c \tag{2.1}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{C}$ is the Coriolis matrix, V is the system potential energy, τ_a is a vector of aerodynamic forces and torques, τ_c is a vector of control forces and torques, and $\boldsymbol{q} = \left[\boldsymbol{q}_1^T, ..., \boldsymbol{q}_L^T\right]^T$ is the entire state vector. Despite the analytical formulation for the nonlinear mass matrix $\boldsymbol{M}(\boldsymbol{q})$ 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 $\mathbf{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ depends on the an-isotropic, internally actuated wing membrane for which only preliminary investigations have been conducted [24]. 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 [92] because unsteady, nonlinear effects are substantial and cannot be neglected [38]. Performing on-line computation of these aerodynamics is

intractable because they require on the order of 10^6 to 10^8 computational degrees of freedom [111, 113] and, consequently, hours or days to complete [112]. 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, Equation 2.1 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 Equation 2.1, 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 Equation 2.1 are poorly understood rendering them effectively useless in this context. Conversely, algorithms which approximate the equations of robotics in Equation 2.1 can be constructed by relying on experimentally determined kinematic variables, $\left\{\boldsymbol{q}_l\right\}_{l=1}^{L}$, 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, Section 2.2 provides a literature review of related research in bat flight and state of the art estimation methods. In Section 2.3, we present an overview of our framework for estimating joint states without a suitable motion model. Next, Section 2.4 discusses the theoretical formulation for our estimator and reviews the GPDM formulation. Section 2.5 presents synthetic simulations and experimental results which show that our estimator out performs traditional methods and that the manifold we identify using GPDM is consistent with the observed motions. Finally, Section 2.6 provides concluding remarks and opportunities for future work.

2.2 Related Research

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 [15, 24, 49, 85, 86, 87, 107, 119]. 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 [15] 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 [21, 44]

or commercially available [105].

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 are prone to systematic, pervasive, and periodic self occlusions [87, 107]. 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 [87]. Additionally, many of the previous studies only use data from a single wing for dimensionality reduction [87], CFD studies [111, 112, 113], and PIV measurements [86, 87, 119].

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 [6] and, therefore, present a formidable challenge to motion studies.

Table 2.1 summarizes the experimental paradigms used in previous studies and provides a contrast to the experimental methods presented in this paper. As shown in Table 2.1,

Source	Cams	fps	Res	Points	Species
[2]	2	200	Film	6	R. ferrumequinum
[107]	2	500	?	≈20	C. brachyotis
[49]	4	200	1024	17	C. brachyotis
[119]	2	250	?	9	G. soricina
[15, 24, 85, 86, 87]	3-4	1000	1024	17-32	C. brachyotis R. aegyptiacus P. pumilus E. hypomelanus P. vampyrus
Our Array (This Paper)	30 (17)	120	720p	≈120 (34)	R. ferrumequinum H. pratti (H. armiger)

Table 2.1: Bat Flight Motion Capture System Specifications

the experimental facility developed in this paper uses an order of magnitude more cameras, operating at $\frac{1}{2}$ to $\frac{1}{8}$ 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 Figure 2.1. 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.

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.

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 [72] discusses the state of the art in human motion capture techniques for monocular, stereo, and redundant camera systems. Another review [34] focuses on motion estimation techniques for the human hand, including the use of various numbers of cameras for pose determination. A more recent review [61] 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 [12, 13, 15]. 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 senor information [106]. One very 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 are performed a priori within each camera view. Previous work conducted by the authors has formulated the motion estimation problem in this fashion [12]. Some work has been done by bootstrapping optical flow or expectation maximization to solve the correspondence problem online [15, 32]. 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 hierarchal tree and represent the distribution as piecewise constant over that tree [97, 98, 104]. Typically the process of partitioning is conducted using a hierarchal 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 [46], the tree structure is used for hierarchal 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 [94]. 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 [65] which is an extension of ICondensation [54] 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.

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 [59, 108, 114, 116]. 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 [108, 114, 116] which develop Gaussian Process Dynamic Models (GDPM) for learning low-order representations of human motions. The GDPM framework extends the work in [59], 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 Section 2.4.

2.3 Overview of Our Method

In this paper we exploit the connectivity of the multibody 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 2.2 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 first two frames are

identical to the estimation scheme presented above: the estimate of the states of links 1 to 4 at time step $k-1$ is depicted in the upper left panel of the figure, and this state is used as the prediction for the next frame. 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, this updated estimate of the 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, a spatially recursive process is derived as 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. In this case, the algorithm starts at the bat body and then proceeds to generate state estimates using measurements of the humerus, radius-ulna, and outboard along the digits. 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 link 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 to 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 Section 2.4 where we present our kinematic representation of bat flight, the derivation of the conditionally independent spatially recursive Bayes' filter 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.

2.4 Theoretical Formulation

In their full complexity, bats are infinite dimensional, nonlinear systems because of their elaborate musculo-skeletal 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 organized into rigid and non-rigid 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.

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 generalized joint coordinates are parameterized using the conventional and well known DH convention [96]. A detailed discussion of the kinematics can be found in [13].

From this kinematic definition, we can assemble state vectors for individual links and groups of links. The state vector for the l^{th} link, $\boldsymbol{q}_l$, contains all degrees of freedom for that link. These state vectors can be grouped using the notation $\boldsymbol{q}_{a:b}$, 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, $\boldsymbol{q}$, to measurements, $\boldsymbol{\phi}$, 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 $\boldsymbol{\phi}_{1:l,1:k}$. This notation is critical in the derivation of the conditionally independent Bayes' filter with inboard state correction (ISC).

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

$$p(\boldsymbol{q}_{1:l,k}|\boldsymbol{\phi}_{1:l,1:k}) = p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{1:l-1,k};\boldsymbol{\phi}_{1:l,1:k})p(\boldsymbol{q}_{1:l-1,k}|\boldsymbol{\phi}_{1:l,1:k}). \tag{2.2}$$

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

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

$$p(\boldsymbol{q}_{1:l,k}|\boldsymbol{\phi}_{1:l,1:k}) = p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{1:l-1,k};\boldsymbol{\phi}_{1:l,1:k})p(\boldsymbol{q}_{1:l-1,k}|\boldsymbol{\phi}_{1:l-1,1:k}). \tag{2.3}$$

We can repeatedly apply the chain rule to $p(\boldsymbol{q}_{1:l-1,k}|\boldsymbol{\phi}_{1:l-1,1:k})$ in Equation 4.16 until it follows that

$$p(\boldsymbol{q}_{1:l,k}|\boldsymbol{\phi}_{1:l,1:k}) = \left(\prod_{l=2}^{L} p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{1:l-1,k};\boldsymbol{\phi}_{1:l,1:k})\right) p(\boldsymbol{q}_{1,k}|\boldsymbol{\phi}_{1,1:k}). \tag{2.4}$$

The second term on the right of Equation 2.4 is the posterior distribution for the joint states of link 1. 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 $p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{1:l-1,k};\boldsymbol{\phi}_{1:l,1:k})$ to complete the spatially recursive filter derivation.

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

$$p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{1:l-1,k};\boldsymbol{\phi}_{1:l,1:k}) = \eta p(\boldsymbol{\phi}_{1:l,k}|\boldsymbol{q}_{1:l,k};\boldsymbol{\phi}_{1:l,1:k-1}) p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{1:l-1,k};\boldsymbol{\phi}_{1:l,1:k-1}). \tag{2.5}$$

Additionally, observations at time k are independent of previous observations given the current, $\boldsymbol{\phi}_{1:l,k} \perp\!\!\!\perp \boldsymbol{\phi}_{1:l,1:k-1}|\boldsymbol{q}_{1:l,k}$. Applying this conditional independence condition

$$p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{1:l-1,k};\boldsymbol{\phi}_{1:l,1:k}) = \eta p(\boldsymbol{\phi}_{1:l,k}|\boldsymbol{q}_{1:l,k}) p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{1:l-1,k};\boldsymbol{\phi}_{1:l,1:k-1}). \tag{2.6}$$

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

$$p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{1:l-1,k};\boldsymbol{\phi}_{1:l,1:k-1}) = \int p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{l,k-1};\boldsymbol{\phi}_{1:l,1:k-1};\boldsymbol{q}_{1:l-1,k})* \\ p(\boldsymbol{q}_{l,k-1}|\boldsymbol{q}_{1:l-1,k};\boldsymbol{\phi}_{1:l,1:k-1}) d\boldsymbol{q}_{l,k-1}. \tag{2.7}$$

Finally, we can apply conditional independence conditions to simplify this expression. For state prediction we employ a random walk motion model—the reason for this is discussed in more detail in the follow section. Due to the random walk motion model and the chain-like structure of the bat skeleton, we can assume that the propagation of link l states are independent of all other states. That is, we have $\boldsymbol{q}_{l,k} \perp\!\!\!\perp \boldsymbol{q}_{1:l-1,k}|\boldsymbol{q}_{l,k-1}$. Also, the propagation of joint states for link l does not depend on previous measurements given the previous states, $\boldsymbol{q}_{l,k} \perp\!\!\!\perp \boldsymbol{\phi}_{1:l,1:k-1}|\boldsymbol{q}_{l,k-1}$. Finally, previous states and future states are independent given current observations, $\boldsymbol{q}_{l,k-1} \perp\!\!\!\perp \boldsymbol{q}_{1:l-1,k}|\boldsymbol{\phi}_{1:l,1:k-1}$. So, we conclude that

$$p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{1:l-1,k};\boldsymbol{\phi}_{1:l,1:k-1}) = \int p(\boldsymbol{q}_{l,k}|\boldsymbol{q}_{l,k-1}) p(\boldsymbol{q}_{l,k-1}|\boldsymbol{\phi}_{1:l,1:k-1}) d\boldsymbol{q}_{l,k-1}. \tag{2.8}$$

Equations 2.3, 2.6, and 2.8 summarize the recursive Bayes' filter with inboard state correction. Equation 2.3 computes the joint probability of all links up to and including link l. Equation 2.6 incorporates new sensor information as well as sensor information from all previously corrected inboard link states. Finally, the prediction is computed using Equation 2.8.

Kalman and Particle filters are often used to implement Bayes' filters. Particle filters are typically 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 analytic 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 DOF for each link are adequately constrained by the sensor data available. However, this is not always true in the 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 [72], and many use random walk for state propagation [12, 23, 43]. Thus, we use this model as well,

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

where $\boldsymbol{w}$ is Gaussian noise that is assumed to have a constant covariance and be zero-mean. While random walk is a very simple motion model it provides two important attributes for our estimation problem. First, it does not assume a direction of motion which is important because we do not want to constrain the estimator to a particular regime of flight. Second, it assumes that states do not change dramatically between adjacent time steps which is common in systems with inertia. Furthermore, the uncertainty injected at each prediction step can be estimated from inspection of the experimental videos. The bat's body is roughly 100mm 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 we approximate as 15mm. 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^o of motion, which occurs over approximately 8 frames. From these observations, we estimate that the wing joint standard deviation is approximately 12^o. Despite these large uncertainties induced by the poor understanding of the bat motion model, 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 [64, 103]. 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 [64]

$$\boldsymbol{\phi} = \begin{Bmatrix} \psi_x \\ \psi_y \end{Bmatrix} = \frac{1}{{}^c\lambda} {}^c\mathbf{K}\boldsymbol{\Pi}_0\mathbf{H}_0^c\boldsymbol{x}^0, \tag{2.9}$$

where λ is distance from the camera focal center to the point $\boldsymbol{x}^0$ in inertial coordinates, ${}^c\mathbf{K}$ is a matrix of camera parameters for the c^{th} camera, $\boldsymbol{\Pi}_0$ is a matrix which maps from homogeneous inertial coordinates to homogeneous pixel coordinates, and $\mathbf{H}_0^c$ is a homogeneous transform which relates the camera frame to the inertial frame. 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 inertial space using the nonlinear kinematics defined in Section 2.4. Thus, the entire sensor model becomes

$$\boldsymbol{\phi} = \begin{Bmatrix} \psi_x \\ \psi_y \end{Bmatrix} = \frac{1}{{}^c\lambda} {}^c\mathbf{K}\boldsymbol{\Pi}_0\mathbf{H}_0^c\mathbf{H}_l^0(\boldsymbol{q}_{1:l})\boldsymbol{x}^l + \boldsymbol{v}, \tag{2.10}$$

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

One source of error in our sensor measurements is due to uncertainty in the calibration parameters calculated by the MultiCamSelfCal [100] toolbox. The reprojection error reported by the toolbox is consistently $\ll 1.5$ 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 [12] 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 Section 2.5. 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.

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 [115]. We provide a review of the foundational theory here which was originally derived for dimensionality reduction of human motion [108, 114, 114].

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

$$\boldsymbol{x}_k = \boldsymbol{f}(\boldsymbol{x}_{k-1}, \mathbf{A}) + \boldsymbol{n}_x, \tag{2.11}$$

$$\boldsymbol{q}_k = \boldsymbol{g}(\boldsymbol{x}_k, \mathbf{B}) + \boldsymbol{n}_y, \tag{2.12}$$

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

$$\boldsymbol{f}(\boldsymbol{x}, \mathbf{A}) = \sum_j \boldsymbol{a}_j \phi(\boldsymbol{x}_j, \boldsymbol{x}) = \mathbf{A}\boldsymbol{\phi}(\boldsymbol{x}), \tag{2.13}$$

$$\boldsymbol{g}(\boldsymbol{x}, \mathbf{B}) = \sum_j \boldsymbol{b}_j \psi(\boldsymbol{x}_j, \boldsymbol{x}) = \mathbf{B}\boldsymbol{\psi}(\boldsymbol{x}), \tag{2.14}$$

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

$$\phi(\boldsymbol{x}_j, \boldsymbol{x}) = \alpha_1 \exp\left(-\frac{\alpha_2}{2}||\boldsymbol{x} - \boldsymbol{x}_j||^2\right) + \alpha_3 \boldsymbol{x}^T \boldsymbol{x}_j + \alpha_4^{-1}\delta(\boldsymbol{x}, \boldsymbol{x}_j), \tag{2.15}$$

$$\psi(\boldsymbol{x}_j, \boldsymbol{x}) = \beta_1 \exp\left(-\frac{\beta_2}{2}||\boldsymbol{x} - \boldsymbol{x}_j||^2\right) + \beta_3^{-1}\delta(\boldsymbol{x}, \boldsymbol{x}_j), \tag{2.16}$$

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 $\mathbf{B}$ [3]. 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 [108, 114, 116] marginalize over the parameters in $\mathbf{A}$ and $\mathbf{B}$ and optimize latent variables and hyper-parameters. To identify these parameters we maximize the joint probability of the collected data $\mathbf{Q}$, corresponding latent states $\mathbf{X} := \begin{bmatrix}\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K\end{bmatrix}$, hyper-parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, and a weighting matrix $\mathbf{W}$ as

$$p(\mathbf{Q}, \mathbf{X}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{W}) = p(\mathbf{Q}|\mathbf{X}, \boldsymbol{\beta}, \mathbf{W})p(\mathbf{X}|\boldsymbol{\alpha})p(\boldsymbol{\alpha})p(\boldsymbol{\beta})p(\mathbf{W}). \tag{2.17}$$

The distribution over the function which maps from latent space to state space is represented by the first term on the right hand side. By assuming a Gaussian on the model parameters in $\mathbf{B}$ they can be marginalized out of the equation. Doing so has two important implications: the model is now independent of the large number of parameters in $\mathbf{B}$, and a closed for expression can be derived [66]. The marginalization results in

$$p(\mathbf{Q}|\mathbf{X},\boldsymbol{\beta},\mathbf{W}) = \frac{|\mathbf{W}|^K}{\sqrt{(2\pi)^{KN}|\mathbf{K}_q|^N}}\exp\left(-\frac{1}{2}\text{trace}\left(\mathbf{K}_q\mathbf{Q}\mathbf{W}^2\mathbf{Q}^T\right)\right), \tag{2.18}$$

where, $\mathbf{W} := \text{diag}(\begin{bmatrix}w_1, w_2, ...w_N\end{bmatrix})$ 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 Equation 2.16, and the matrix is assembled as

$$(\mathbf{K}_q)_{ij} = \beta_1\exp\left(-\frac{\beta_2}{2}||\boldsymbol{x}_i - \boldsymbol{x}_j||^2\right) + \beta_3^{-1}\delta(\boldsymbol{x}_i, \boldsymbol{x}_j), \tag{2.19}$$

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 Equation 2.17 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

$$p(\mathbf{X}|\boldsymbol{\alpha}) = \int p(\mathbf{X}|\mathbf{A},\boldsymbol{\alpha})p(\mathbf{A}|\boldsymbol{\alpha})d\mathbf{A}. \tag{2.20}$$

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

$$p(\mathbf{X}|\boldsymbol{\alpha}) = \frac{p(\boldsymbol{x}_1)}{\sqrt{(2\pi)^{(K-1)D}|\mathbf{K}_x|^D}}\exp\left(-\frac{1}{2}\text{trace}\left(\mathbf{K}_x^{-1}\mathbf{X}_{2:K}\mathbf{X}_{2:K}^T\right)\right). \tag{2.21}$$

In this equation, the latent variables are non-dimensional 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 Equation 2.15 and is assembled as

$$(\mathbf{K}_x)_{ij} = \alpha_1\exp\left(-\frac{\alpha_2}{2}||\boldsymbol{x}_i - \boldsymbol{x}_j||^2\right) + \alpha_3\boldsymbol{x}_i^T\boldsymbol{x}_j + \alpha_4^{-1}\delta(\boldsymbol{x}_i, \boldsymbol{x}_j), \tag{2.22}$$

for all $i, j \in \begin{bmatrix}2, \ldots, K\end{bmatrix}$. The next terms in the distribution in Equation 2.17 are the priors

on $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ which are formulated as

$$p(\boldsymbol{\alpha}) = \prod_i \alpha_i^{-1} \tag{2.23}$$

$$p(\boldsymbol{\beta}) = \prod_i \beta_i^{-1} \tag{2.24}$$

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

The final term in Equation 2.17 is the prior on the weighting matrix. For this paper, a half normal prior is assumed on each entry in $\mathbf{W}$

$$p(\mathbf{W}) = \prod_{m=1}^{D} \frac{2}{\kappa\sqrt{2\pi}} \exp\left(-\frac{w_m^2}{2\kappa^2}\right), \tag{2.25}$$

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 [116].

The probability distribution in Equation 2.17 can be maximized by minimizing the negative log likelihood of the distribution which is expressed as

$$\begin{aligned} L = \frac{D}{2}\ln||\mathbf{K}_x|| + \frac{N}{2}\ln||\mathbf{K}_q|| - K\ln|\mathbf{W}| + \frac{1}{2}\text{trace}(\mathbf{K}_x^{-1}\mathbf{X}_{2:K}\mathbf{X}_{2:K}^T) + \frac{1}{2}\boldsymbol{x}_1^T\boldsymbol{x}_1 + \\ \frac{1}{2}\text{trace}(\mathbf{K}_q^{-1}\mathbf{Q}\mathbf{W}^2\mathbf{Q}^T) + \sum_j \ln(\beta_j) + \frac{1}{2\kappa^2}\text{trace}(\mathbf{W}^2) + \sum_j \ln\alpha_j. \end{aligned} \tag{2.26}$$

To minimize the loss function above, we use Balanced GPDM as in [115]. This approach iterates multiple times over two phases. The first phase holds hyper-parameters fixed while optimizing latent states. The second phase holds the latent states fixed while optimizing hyper parameters. This process iterates until a maximum number of cycles elapses or a set threshold is reached.

Once the optimization completes, trajectories can be propagated in latent space using the flowing distribution

$$\boldsymbol{x}_k \sim \mathcal{N}\left(\mu_X(\boldsymbol{x}_{k-1}); \sigma_X^2(\boldsymbol{x}_{k-1})\mathbf{I}\right), \tag{2.27}$$

where μ_X is the mean function and σ_X^2 is the covariance function. Due to the construction of the posterior distribution, the mean and covariance functions are defined as

$$\mu_X(\boldsymbol{x}) = \mathbf{X}_{2:K}^T\mathbf{K}_x^{-1}\boldsymbol{k}_X(\boldsymbol{x}), \tag{2.28}$$

$$\sigma_X^2(\boldsymbol{x}) = \phi(\boldsymbol{x},\boldsymbol{x}) - \boldsymbol{k}_X(\boldsymbol{x})^T\mathbf{K}_x^{-1}\boldsymbol{k}_X(\boldsymbol{x}). \tag{2.29}$$

where $\boldsymbol{k}_X(\boldsymbol{x}) = \left[\phi(\boldsymbol{x}_2, \boldsymbol{x}), \ldots, \phi(\boldsymbol{x}_K, \boldsymbol{x})\right]$. To project latent trajectories back into state space the mean and covariance functions can be written as

$$\mu_Q(\boldsymbol{x}) = \mathbf{Q}^T \mathbf{K}_q^{-1} \boldsymbol{k}_Q(\boldsymbol{x}), \tag{2.30}$$

$$\sigma_Q^2(\boldsymbol{x}) = \psi(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}_Q(\boldsymbol{x})^T \mathbf{K}_q^{-1} \boldsymbol{k}_Q(\boldsymbol{x}), \tag{2.31}$$

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

2.5 Results

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 Section 2.4. 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 2.3 - 2.5 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 Figure 2.3, 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 Figure 2.4, 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 2.5 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 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 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 2.6 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 120fps, dashed lines represent 240fps, and dotted lines represent 480fps. At a frame rate of 120fps, 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 480fps. 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.

Experimental Estimation Results

The conditionally independent UKF with inboard state correction which is derived above is applied to estimate joint states of bats in flight. The video data was collected at the SDU-VT International Laboratory at Shandong University in Jinan, China. The array of cameras used to collect raw video sequences of bats in flight consisted of 21 GoPro Hero 3+ action cameras recording 720p video at 120 frames-per-second. A male Hipposiderous pratti was used in the experimental data presented in this paper. The estimator derived above is used to estimate the joint states of the bat and reconstruct the motion of 34 marker points distributed across the bat's body and wings. All link names, DH parameters, and estimation group assignments are included in a previous paper [13].

Prior to conducting motion capture experiments, the system was calibrated using the MultiCamSelCal Toolbox [100]. 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 though 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 2pixels 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 nonrigid 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 2.7 - 2.9 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, dashed lines represent the CI estimator, and dotted lines represent the standard UKF. Note that in the first group of states which are shown in Figure 2.7, the CI and ISC estimators return comparable estimates while the standard estimator contains significantly more noise and diverges from the other two estimates.

In Figure 2.8, the CI and ISC estimator perform comparably well for all states. The standard estimator deviates substantially from the other two estimates. Figure 2.9 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 2.10 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 Figure 2.11. In the figure, ISC UKF reprojection errors are denoted by the '+' markers and CI UKF errors are denoted by the 'o' markers. 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 ±50pixels. 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.

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. Excluding the body motion ensures that the identified model is independent of the a particular path of the bat. To identify a model for straight and level flapping flight we ran the optimization described above on 80 time steps of flapping flight data. In addition to the experimental data, we specified the number of latent degrees-of-freedom as $D = 2, 3,$ or 4 and initialized the latent states by using the first D modes of a PCA decomposition of the original kinematics data.

Figure 2.12 shows the resulting manifolds. Subfigures (a), (b), and (c) show the manifolds identified when assuming that the latent space has two, three, and four DOF, respectively. The identified latent states are represented by the blue points while the green points indicate fair HMC samples around these latent states. The purpose of the HMC samples is to demonstrate uncertainty in the manifold. Finally, the red trajectory is a simulation which extrapolates samples beyond the experimental data used for optimization. If the bat performed each flap cycle the same, the blue points would appear to be periodic repetitions, the green points would show little deviation from the identified trajectories, and the red points would continue to trace the original latent states identified in the optimization. While none of the figures show the ideal behavior important information can be gleaned from this study.

The manifold which assumes 2 latent DOF is shown in Figure 2.12a. The latent states exhibit large jumps in the trajectory and there is little consistency between cycles. Figure 2.12b which assumed three latent DOF shows a manifold which is smooth and exhibits little uncertainty. Despite the small uncertainty, only semi-periodic motion is observed. Further discussion of this feature is provided below. Finally, the 4 DOF latent space illustrated in 2.12c. This manifold has high uncertainty in the third DOF and acyclic motions are observed. Thus, 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.

To provide an intuitive interpretation of the identified 3 DOF manifold, Figure 2.13 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 Figure 2.13 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 down stroke begins at the yellow circle. Finally, the magenta circle corresponds to the middle of the down stroke and the blue circle to the bottom of the down stroke. 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, Figure 2.14 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, become 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.

The projection of latent states into state space is shown in Figure 2.15. The solid and dashed lines which represent the predictions match the experimental data represented by dots. Note that for $\theta_1 - \theta_3$ the amplitude of oscillation is larger at the beginning of the test (solid line) than at the end (dashed line). This behavior is consistent with the previous discussion of the V_3 dimension and may explain the semi-periodic shape of the manifold shown previously. Additionally, the extrapolated predictions (dotted line) appear to be periodic repetitions of the experimental data.

While the consistency of the projections from latent space into feature space is encouraging, we aim to understand the cause of the aperiodic nature of motion in the V_3 dimension of the latent space. As discussed previously, if all flap cycles in the experiment where kinematically equivalent, the latent space should show periodic repetitions of the same motion. Because of the semi-periodic nature of the data, we suspect that there exist important variations in flap cycle for this test. By computing the linear velocity of the bat's body, which is shown in Figure 2.16, it is apparent that the bat was accelerating for the first half of the experiment and maintaining constant velocity for the remainder of the flight. In Figure 2.16 the sections of the manifold which correspond to acceleration and constant velocity are correlated to the velocity plot using similar line types. Solid lines correspond to acceleration and dashed lines correspond to the constant velocity portion of the test. From this figure, it is clear that there are two different segments to the manifold which is consistent with the analysis presented previously. Thus, we have successfully identified a discrete dynamic model for straight and level bat flight.

2.6 Conclusions

The first goal of this paper is to estimate the position and pose of a bat during a straight and level flight maneuver. While this type of work has been conducted previously, the low-resolution, low-frame rate cameras which we employ in this paper make standard state estimation and marker tracking methods fail. To overcome these issues we formulate a conditionally independent Bayes' filter which incorporates inboard state correction (ISC) for trajectory estimation. 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 where 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 identify a low dimensional representation of bat flight. Our results demonstrate that a three-dimensional manifold is sufficient for describing straight and level flight of bats. In addition, we identify a dynamic model within this latent space and a projection from latent space to feature space such that the model can be used as a motion prior for future experimental studies. To the best of our knowledge, this is the first nonlinear dimensionality reduction of bat flight kinematics.

Figure 2.1: Point occlusions are frequent and periodic in bat flight. The top camera captures the down stroke well, but occlusion occurs on the upstroke. Conversely, the side camera captures the upstroke well, but portions of the down stroke are occluded.

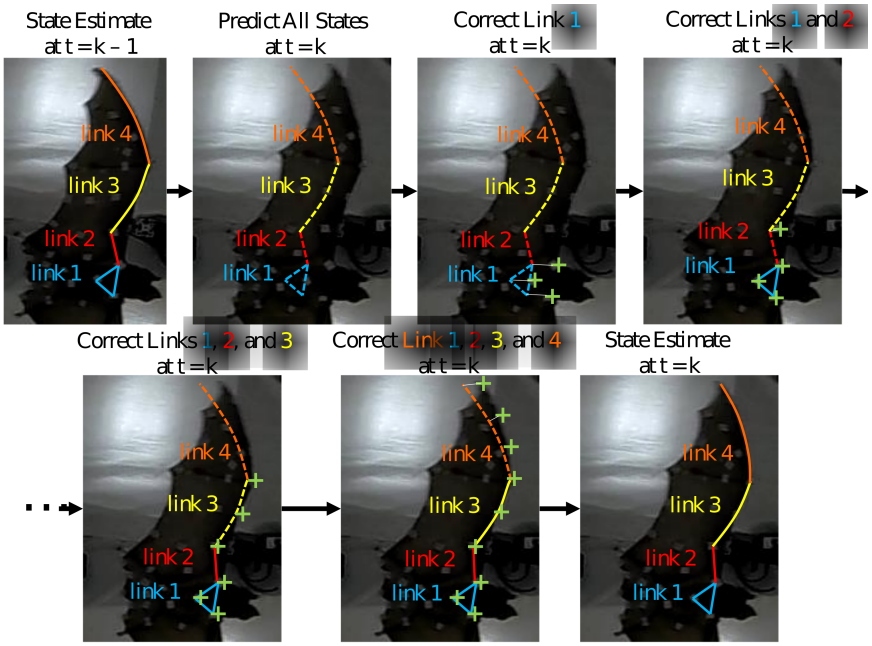


Figure 2.2: Conditionally Independent Spatially Recursive Bayesian Estimation. Unlike the i.i.d. approach to recursive estimation, the conditionally independent estimator corrects all previous link states as new link states are added. This prevents over fitting noise for root links which improves estimation accuracy.

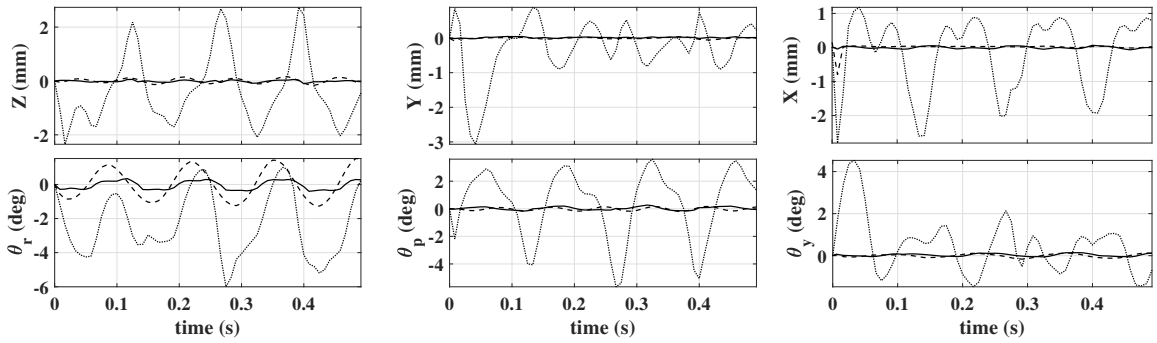


Figure 2.3: Estimation Error of Group 1. Both the CI (dashed line) and the ISC (solid line) estimators perform equally well for the these states while the standard estimator (dotted line) deviates markedly from the ground truth state.

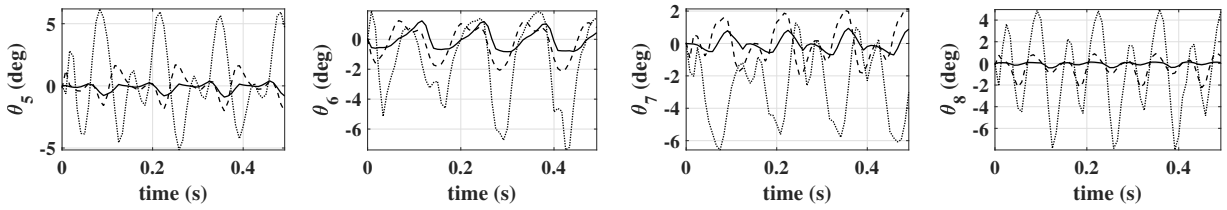


Figure 2.4: Estimation Error of Group 2. The ISC estimator (solid line) starts to out perform the CI estimator (dashed line). The standard estimator (dotted line) still exhibits large errors in the state estimates.

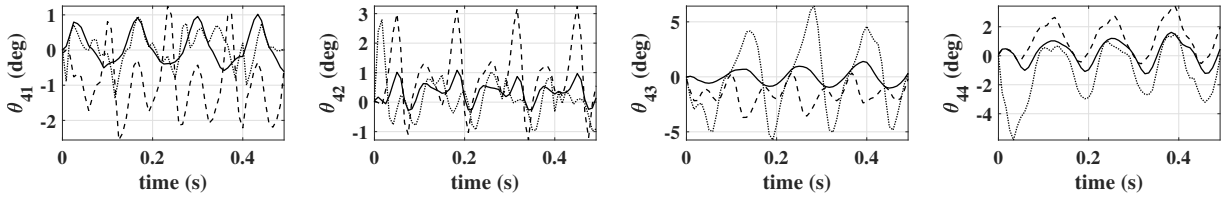


Figure 2.5: Estimation Error of Group 5. The ISC estimator (solid line) is clearly out performing the CI (dashed line) and standard (dotted line) estimators. The image space motion of the features on these links is highly nonlinear due to the kinematics of the bat skeleton.

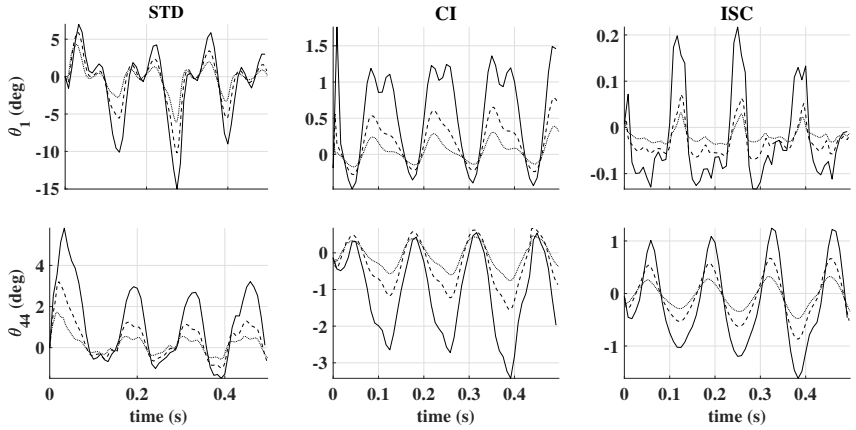


Figure 2.6: Effect of Sample Rate on Estimated States for Synthetic Data. The sample rates are: 120fps (solid), 240fps (dashed), and 480fps (dotted). Top row shows θ_1 and bottom row shows θ_{44}. From left to right: standard, CI, and ISC error estimates.

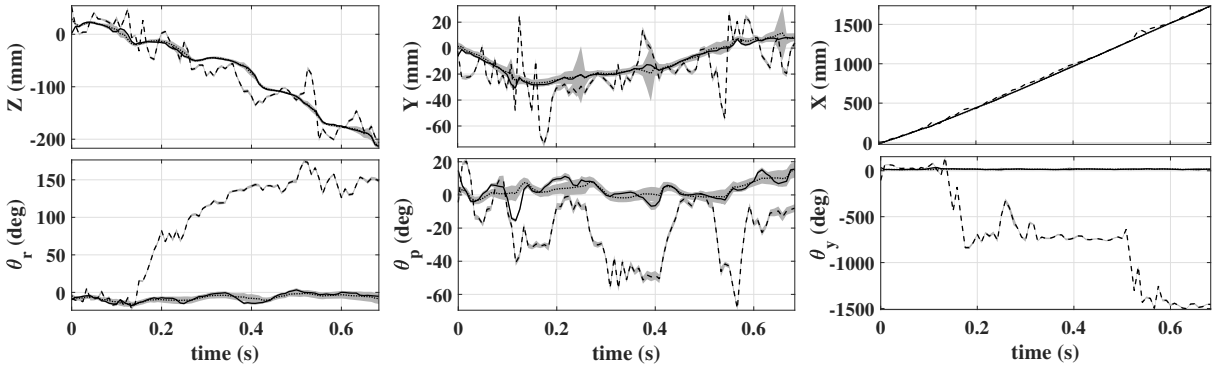


Figure 2.7: Group 1 State Estimates. Top Row: Z, Y, and X body displacement. Bottom Row: Roll, Pitch, and Yaw angles. The ISC (solid) and CI (dotted) estimates are almost equivalent. Large amounts of high frequency noise are present in the standard estimator (dashed). Gray shaded regions represent $\pm 2\sigma$ deviation.

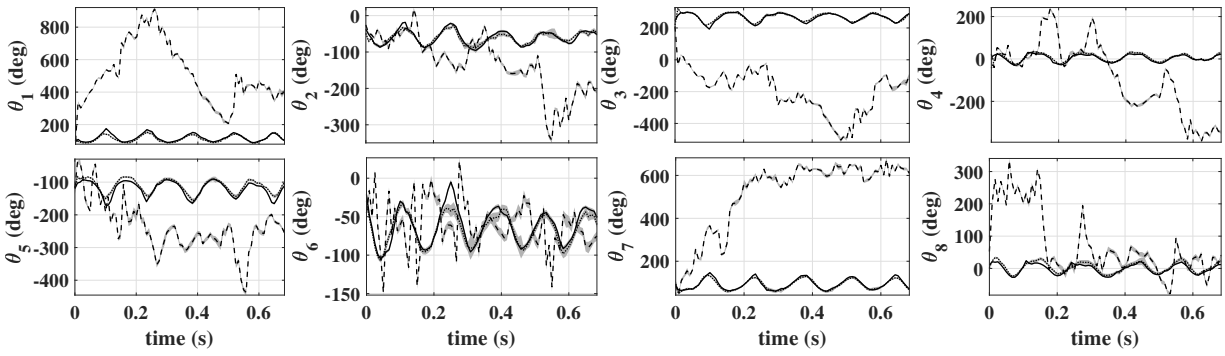


Figure 2.8: Group 2 State Estimates. θ_1 - θ_4: right arm joint angles. $\theta_5 - \theta_8$: left arm joint angles. ISC Method (solid), Standard (dashed), and CI (dotted) estimates. Gray shaded regions represent $\pm 2\sigma$ deviation.

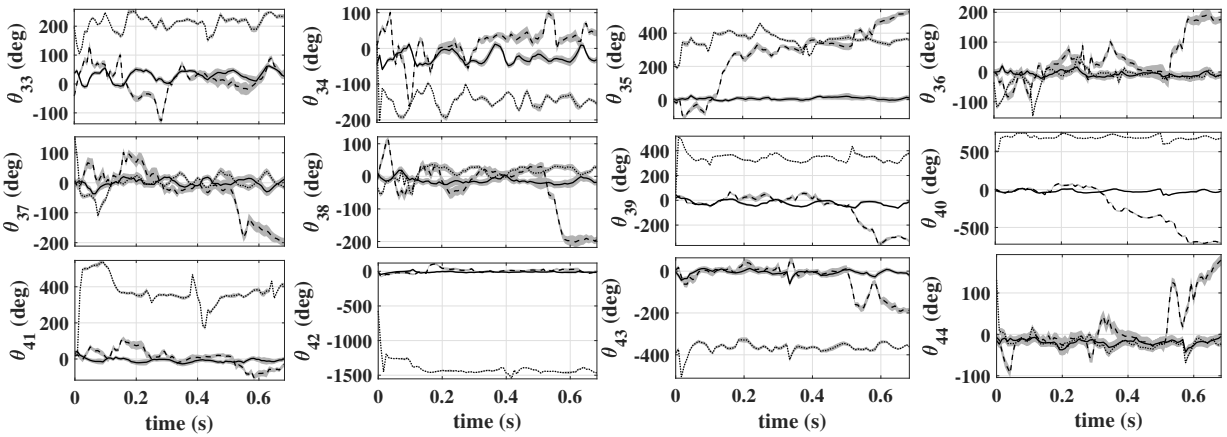
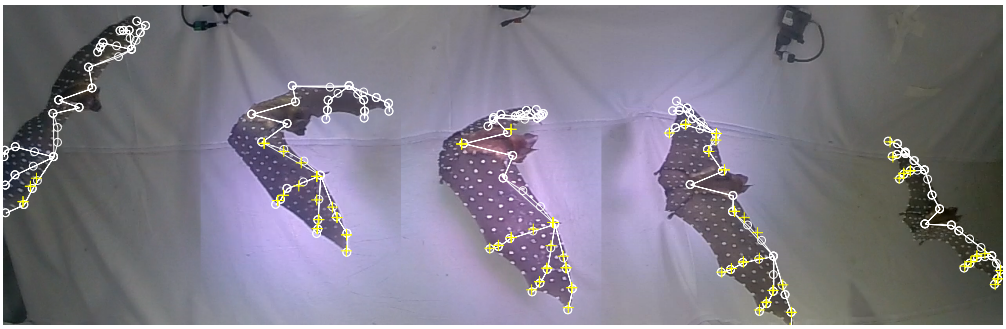
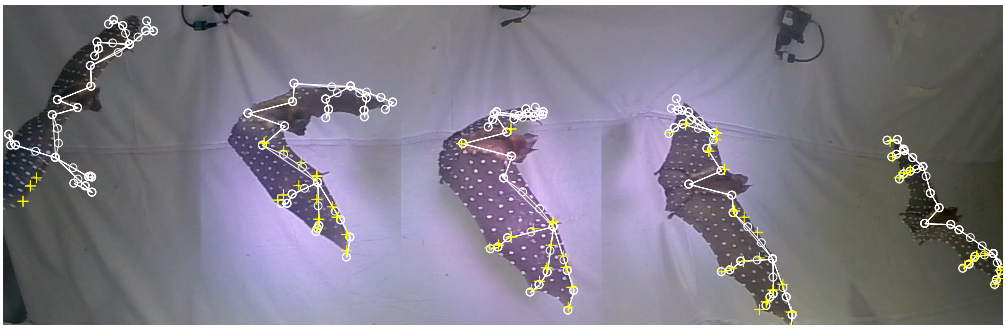


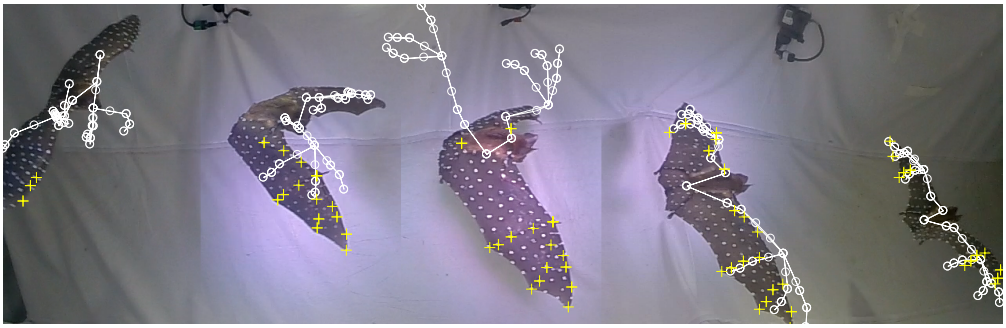
Figure 2.9: Group 5 State Estimates. θ_{33} - θ_{38}: Left hand second phalanx joint angles. θ_{39} - θ_{44}: right hand second phalanx joint angles. ISC (solid), Standard (dashed), and CI (dotted) estimates. Gray shaded regions represent $\pm 2\sigma$ deviation.



(a) ISC

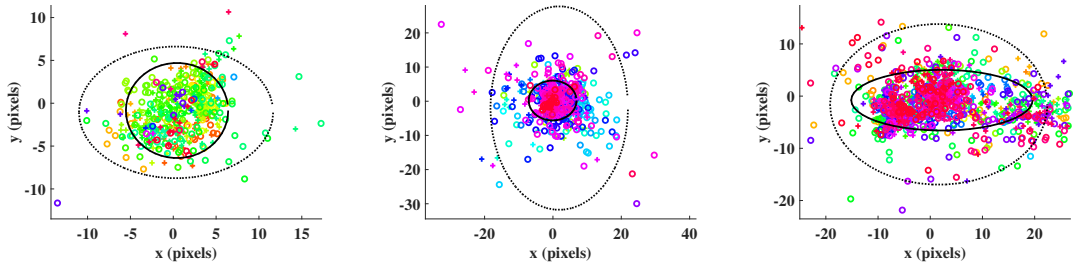


(b) CI



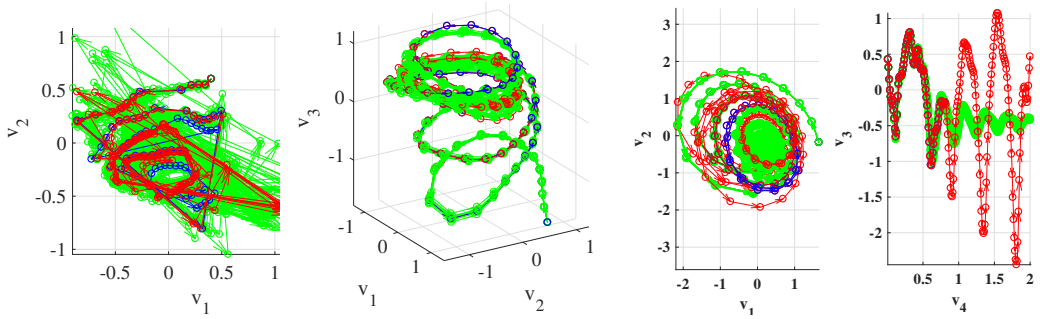
(c) STD

Figure 2.10: 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.



(a) Camera 1 (b) Camera 2 (c) Camera 3

Figure 2.11: 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.



(a) 2 Lat DOF (b) 3 Lat DOF (c) 4 Lat DOF

Figure 2.12: 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.

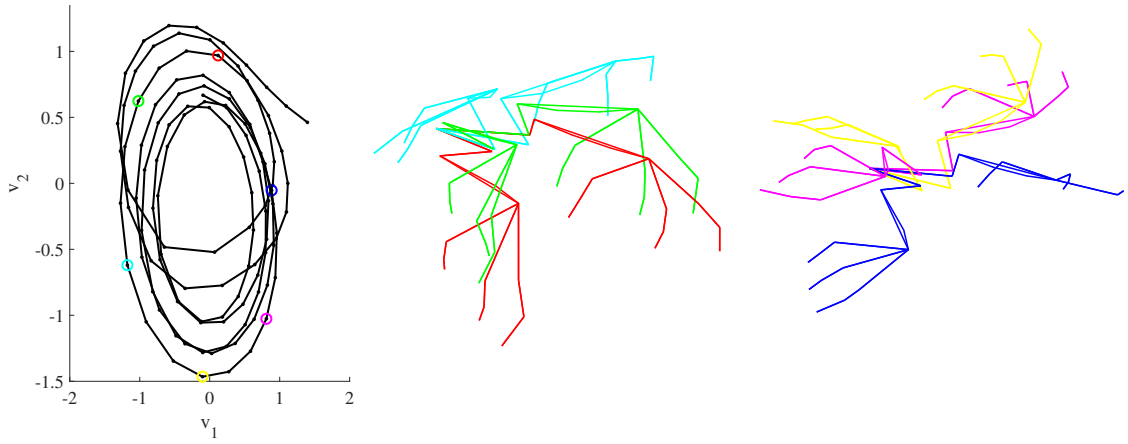


Figure 2.13: Understanding the Latent Space: V_1 and V_2. The identified manifold is depicted at the left. The circle marks are color corresponded to the skeletons in the center and left figures. Starting at the red circle, the wings are approximately at the bottom of the upstroke. The upstroke continues counter clockwise around the quasi-circular manifold. The top of the upstroke is indicated by the yellow marker, and the bottom of the down stroke 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.

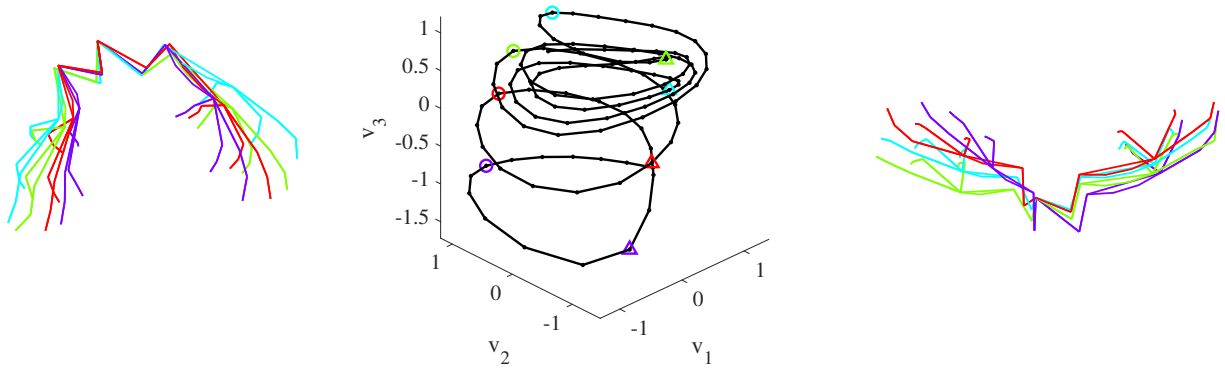


Figure 2.14: Understanding the Latent Space: V_3. In the center figure, circle markers are placed at latent states which correspond to the bottom of the down stroke and triangle markers are placed at locations which correspond to the top of the down stroke. As the markers get lighter in color, V_3 increases. 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.

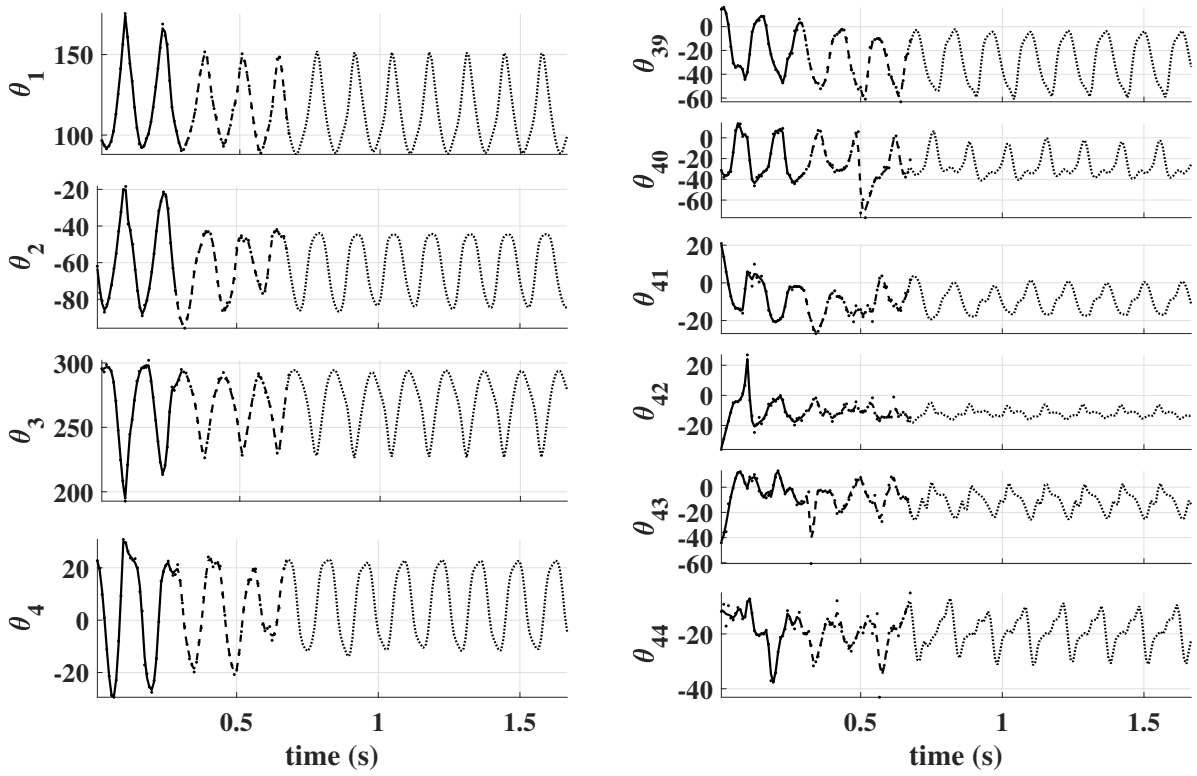


Figure 2.15: Feature Space Projection of Latent Trajectories. θ_1 through θ_4 are shoulder and elbow rotations for the right wing. $\theta_{39} - \theta_{44}$ are second phalanx rotations on the right hand. Dots represent the training data points. The first two flap cycles (solid lines) are of notably 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.

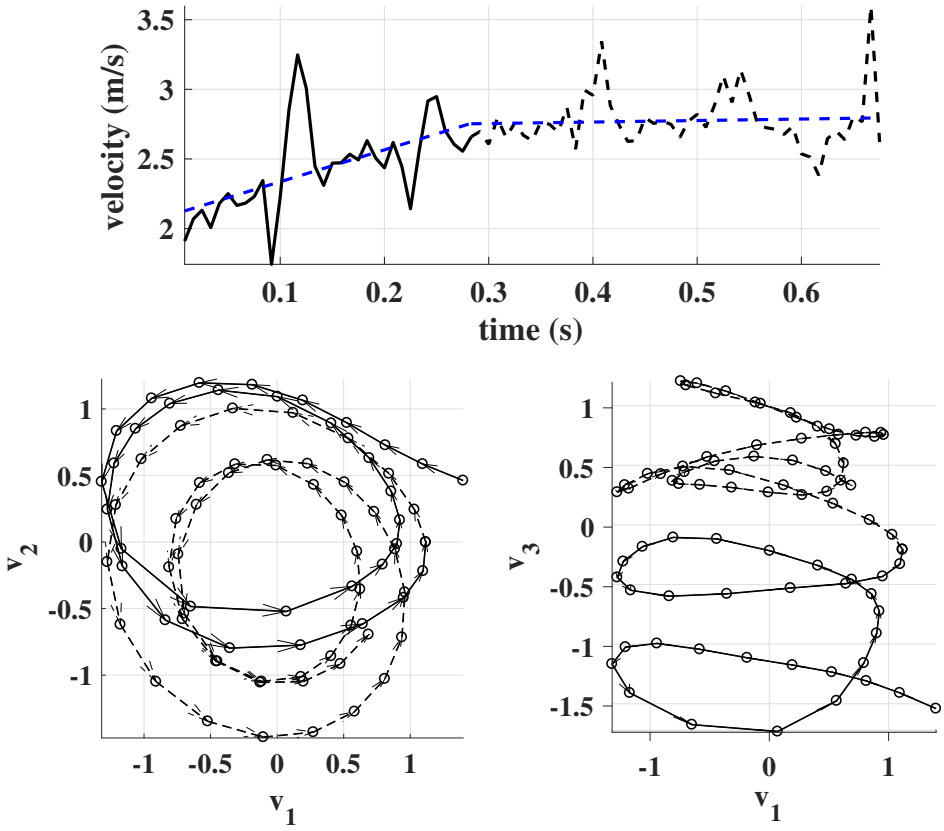


Figure 2.16: [Left] Magnitude of Body Velocity. [Middle and Right] Two views of the manifold. The bat is clearly accelerating (solid lines) for the first half of the test and coasting (dashed lines) for the remainder. These line types correspond to the line types used in Figure 2.15.

Chapter 3

Empirical Potential Functions for Driving Bioinspired Joint Design

Bioinspired design of robotic systems can offer many potential advantages in comparison to traditional architectures including improved adaptability, maneuverability, or efficiency. Substantial progress has been made in the design and fabrication of bioinspired systems. While many of these systems are bioinspired at a system architecture level, the design of linkage connections often assumes that motion is well approximated by ideal joints subject to designer-specified box constraints. However, such constraints can allow a robot to achieve unnatural and potentially unstable configurations. In contrast, this paper develops a methodology which identifies the set of admissible configurations from experimental observations and optimizes a compliant structure around the joint such that motions evolve on or close to the observed configuration set. This approach formulates an analytical-empirical potential energy field which "pushes" system trajectories towards the set of observations. Then, the strain energy of a compliant structure is optimized to approximate this energy field. While our approach requires that kinematics of a joint be specified by a designer, the optimized compliant structure enforces constraints on joint motion without requiring an explicit definition of box-constraints. To validate our approach we construct a 1-DOF elbow joint which closely matches the analytical-empirical and optimal potential energy functions and admissible motions remain within the observation set.

3.1 Introduction

Bioinspiration has become an ubiquitous theme in the design of novel robotic systems due to the adaptability, efficiency, and maneuverability with which biological systems interact with their environment. Humanoid robots and flapping wing micro air vehicles are two excellent examples of how bioinspiration [56, 77] can lead to novel robotic designs. Such robotic systems are often modeled as collections of rigid or flexible bodies connected by ideal joints, as studied in numerous texts on robotics [96] or more generally multibody dynamics [35]. More recent examples of bioinspiration, such as robotic fish and humanoid inspired hand manipulators, incorporate compliant joints to produce more realistic bioinspired robots and to reduce the number of actuators required for practical control designs [9, 62]. To date, the compliant and conventionally designed robotic systems inspired by biomechanical models rely

heavily on designer interpretation of biomotion studies. The output of these experiments may be an intuitive and qualitative understanding of animal motion or it may be a quantitative numerical model generated from videos or images of the specimen in motion. In either case, the observations are subsequently used by an analyst to propose a low dimensional surrogate robot that can emulate the desired motion regimes. In contrast, this paper presents an analytical framework which learns a motion submanifold from experimental observations of motion and optimizes a compliant structure around a joint such that motions evolve on or near the identified manifold.

The methodology presented in this paper relies on experimental observations of biological motion. Detailed discussions of the motion capture experiments and methods for generating thee-dimensional motion data from multiview video sequences of bat flight are reported in [12, 13, 14]. Sample results are shown in Figure 3.1 which depict the estimated skeleton position and pose (white lines and circles) reprojected onto the original image. The yellow points are the original image features which

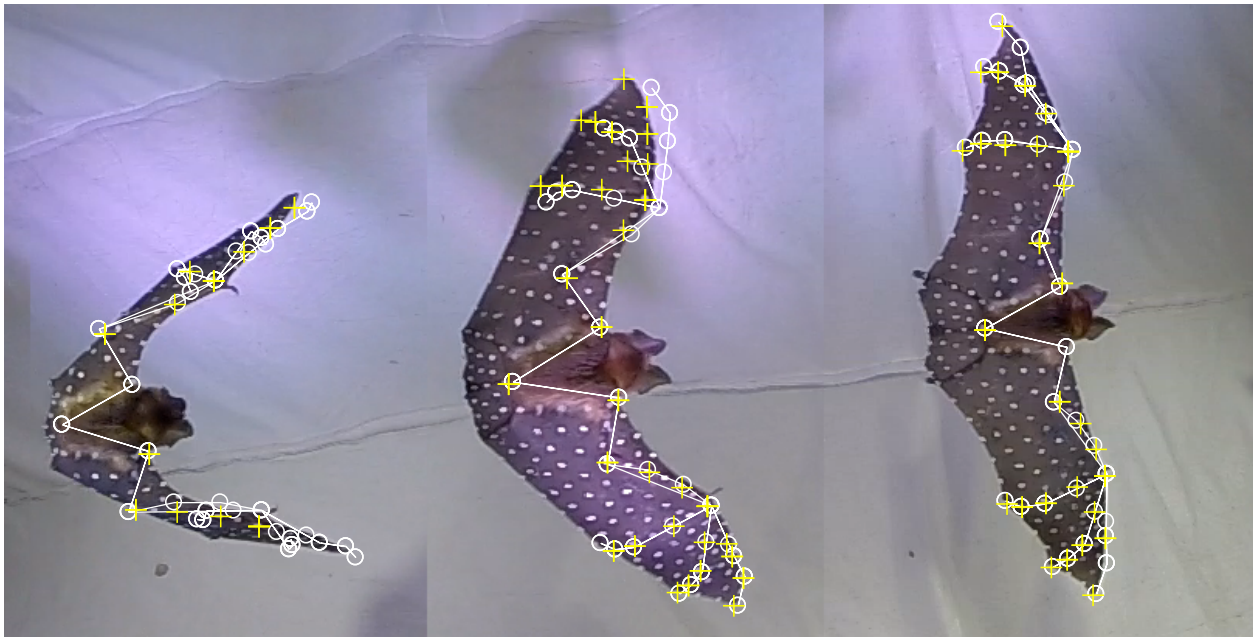


Figure 3.1: Estimated Skeleton Reprojected into a Camera View. White lines and points are the skeletal reprojections. Yellow '+' marks are the original image features.

are used as measurements during state estimation. Because the estimators reported in [13, 14] are formulated in joint space, joint motion is isolated by using forward kinematics [96]. For this paper, we express the motion of the wrist relative to the humerus-fixed frame. We are only interested in the motion of the wrist point due to flexion and extension of the elbow. This data is used to construct an analytical-empirical potential energy function which restricts motions to those observed during the experiment.

Essentially, the analytical-empirical potential characterizes a motion submanifold on which

configurations evolve. After constructing this model and specifying the general joint architecture, a compliant structure is optimized around the joint which ensures that configurations evolve on, or at least close to, the identified manifold. Our approach requires that a designer choose the joint type, but the structure which restricts the joint to the reachable set observed during experiment is learned from data. To the best of our knowledge, this is the first attempt at using biological motion capture to directly design compliant joints: all of the previous theories rely expert interpretation of the observed motion to determine joint limits and design the geometry to ensure that motions stay within the set of observations. To properly frame our work with respect to the existing literature, we begin with a review of current work in biological motion capture, conventional robotic design, compliant mechanisms, and under-actuated robotic design. We then outline our approach which uses learning theory for development of analytical-empirical models.

3.1.1 Literature Review

In the last 20 years, much progress has been made in understanding and modeling animal motion. Perhaps the most mature example is the study of human motion. Techniques for capturing and identifying motion of humans range from traditional marker-based motion capture, for which there are a few commercially available platforms such as VICON™and OptiTrack™, to markerless motion capture systems like the Microsoft Kinect™. Developments in image processing and machine learning have enabled strike-a-pose recognition in single image frames without the need for temporal tracking [33]. The number of articles devoted to this topic is large and cannot be extensively reviewed in this short paper. A seminal survey of these methods and many others used to study human motion is presented in [72]. Because of the substantial investment of research efforts in this area, benchmark data sets of human motion exist [40, 93] which enable researchers to study many classes of motion without repeating time-consuming and expensive experiments. These data sets have been used to study musculo-skeletal pathologies, design more efficient prosthesis and exoskeletons, plan humanoid robot joint trajectories, and design humanoid robotic hardware. This last topic is the most relevant to our work.

Particularly germane to the methods presented in this paper is the design of compliant anthropomorphic hand manipulators. Early hand manipulators were often designed using conventional articulated links, passive springs, and actively controlled actuators. Current trends focus more on incorporating compliant joints to achieve articulations. The authors of [62, 63] design a robotic finger and hand using compliant spring elements inside the joints instead of conventional pins. They note that the compliance built into the joint resolves the tendon tensioning issues that arise from using pulleys to transmit joint torques. Omitting the bulky pulleys from the joints also enables the embedding of sensors within the joints to determine position and external contact forces. While this work employs compliant connections, joint design and the admissible configuration space of the fingers is largely determined by designer interpretation of biological motion capture experiments.

In addition to the study of humanoid robots, a large body of work was conducted in the early to mid 2000s to study the complex kinematics, dynamics, and aerodynamics of bat flapping flight. The seminal body of work conducted at Brown University exemplified that the bat is an exceptional flier in the animal kingdom. This body of work and associated investigations included kinematic analysis [15, 16, 24, 52, 53], experimental aerodynamics studies [48, 49] and dimensionality reduction of complex kinematics [6, 87]. These efforts enabled the design and construction of some early biomimetic prototypes [25]. Using this impressive body of work, researchers at Caltech have created a free-flying bat robot documented in [81, 82, 83]. While the progression from motion capture experiments to a free-flying bat robot is an excellent example of bioinspiration from experimental data, selecting joint ranges of motion and design of the hardware is driven by expert interpretation of the experimental data and dimensionality reduction studies.

Additionally, similar research has been conducted with swimming robots inspired by the motion of the bluegill sunfish. Researchers in [36] present detailed kinematics of the swimming motion of these animals which was used to create the fish inspired robot in [9]. The authors note in this study that the stroke can be segmented into two phases: propulsion and recovery. Thus, in this initial work, the pectoral fins only contained one degree of freedom which swept a fin in the posterior direction for propulsion and in the anterior direction for recovery. The authors note that the recovery portion of the stroke significantly reduced the body velocity of the robot. Thus, in [7] the authors add a passive degree of freedom which enables the fin to pronate and supinate so that the cross sectional area of the pectoral fin is minimized during recovery and maximized during propulsion. Again, the joint is design, both the specification of kinematics and admissible range of motion, is driven by expert interpretation of the results of experimental biomotion studies.

In contrast to bioinspired robot design which typically relies on analyst interpretation of kinematics data, there has been significant progress in using structural optimization to generate 3D printable action figures from characters modeled by animators. The goal of this body of work is to reduce the time and talent required to generate poseable 3D figures from the skinned or solid meshes that are used for animation. A few authors have automated the design of ball joints for articulated characters such that they can be statically posed [5]. Additionally, the authors of [79] optimize a hexagonal wire frame so that certain deformed shapes can be achieved by applying different sets of static input forces. The authors choose a few static configurations and optimize the geometric parameters of the character based on a cost function which penalizes the deviation of the printed character's deformation from the desired pose. This optimization is conducted subject to force equilibrium constraints. While the authors use strain energy [17] to formulate the constraints, they do not optimize directly over the energy function. Rather, they differentiate the energy function to determine internal forces and require that the internal and external forces sum to zero—as would be expected for a system in static equilibrium.

Another approach in this field [95] modifies material properties within the volume of a three-dimensional character to achieve some desired poses when external loads are applied. The

authors generate a CAD model of a three-dimensional character and export the meshed solid in undeformed and deformed configurations. This approach formulates the strain between the undeformed mesh and the deformed mesh as the ratio of characteristic tetrahedral elements within the meshed volume. The authors also use the discrete elastic rod method developed in [17] to compute the strain energy density within the model. Again, the strain energy is differentiated to compute the internal forces in the object and the optimization is conducted subject to static equilibrium constraints. In this paper, we develop a new approach which uses motion capture data of a bat in flight to directly design a compliant structure around a designer-specified kinematic mechanism which is physically realizable. The next section outlines our approach for designing the compliant structure.

3.1.2 Our Approach

The methods presented in this paper differ from previous bioinspired robotics and poseable character studies in two fundamental ways: (1) direct observations of motion in the full configuration space are used to identify a motion submanifold which approximates these observations, and (2) an optimization problem is formulated by minimizing the difference between an empirical potential energy function and the physical strain energy of our selected joint geometry. Firstly, our method optimizes joint geometry directly from the experimental data by identifying an empirical potential energy field. This energy field "pushes" system configurations toward the set or submanifold we observed during motion capture experiments. The advantage of this approach is that designer interpretation is limited to the selection of a general joint architecture and a compliant structure is optimized around the joint to mimic the identified energy field over the configuration space of the joint. One such architecture could be a pin or ball joint with compliant elements which penalize joint motions away from a set of observed configurations. In this paper, we choose a pin joint to approximate elbow motion of a flying bat. Secondly, our work differs from the previous poseable character studies in the formulation of the optimization problem. Our cost function does not penalize deviations from static target poses, but rather penalizes the difference in energy over the entire range of motion. Furthermore, we do not subject the optimization to force constraints, but rather enforce box constraints on the geometric parameters such that our optimization yields geometry which is physically realizable. Our approach is summarized in four general steps: (1) collect experimental observations of the system of interest, (2) construct an analytical-empirical energy field over the observed joint configurations, (3) choose a suitable joint structure to optimize, and (4) optimize the joint structure such that the difference in strain energy of the joint and the identified analytical-empirical energy is minimized.

In this paper, we use experimental observations of bat flight as reported in [13, 14] to design a compliant realization of a bat elbow joint for a bat robot. We choose to approximate the elbow motion of a bat using a 1-DOF pin joint. We construct the analytical-empirical potential using methods described in [28]. Finally, we optimize geometric parameters which

appear in the analytical-empirical model to approximately match the energy of the observed motions. An example image from our experiments and the optimized joint geometry are depicted in Figure 3.2. Before presenting the analytical methods for designing joint geometry, we conclude this section with a brief introduction to analytical-empirical models.

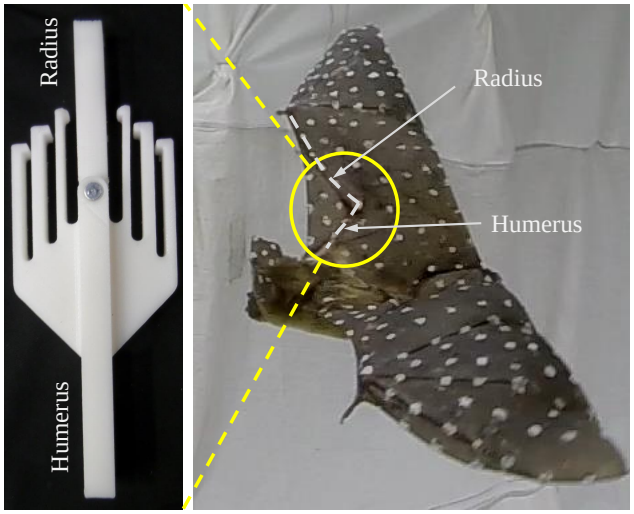


Figure 3.2: Experimental Image (right) and Example Compliant Joint (left). We isolate the elbow motion from our kinematics data and optimize the geometry of the compliant joint.

3.1.3 Analytical-Empirical Models

For many years, the derivation of models for classical mechanical systems such as ground vehicles, air vehicles, marine vehicles, robotic systems, satellites, space stations, gear trains, mechanisms, ... have been taught in texts on analytical mechanics as exemplified by [37, 39, 69]. The last few decades has seen the further development of these approaches in rigorous studies of geometric mechanics [1, 19, 22]. One particularly important application of the general theory of geometric mechanics has been in the development of corresponding theories for discrete Lagrangian or Hamiltonian mechanics as summarized in [68, 73]. These approaches have proven vital in developing discrete integration laws as in [41] that preserve important system invariants such as linear momemntum, angular momentum, or total mechanical energy for a classical mechanical system.

Recently in [28] the authors have investigated a theoretical framework for deriving discrete Lagrangian or Hamiltonian models that in some approximate sense preserve an underlying subset or submanifold that supports the dynamics of a complex system. This approach gives a means of creating 'analytical-empirical' (AE) models that exploit any known geometric structure of the system with information obtained from direct experimental observation of the mechanics of the problem at hand. The approach in [28] is intended to generate analytical-empirical models of biological motion, such as those considered in the study of

human motion [42] or bat motion [13, 14]. The critical contribution of the work in [28] has been to use recent machine-learning approaches [31, 88], along with observations of the biosystem under study to determine an empirical function that essentially drives the evolution of a high dimensional model so that it approximately lies on an empirically-determined, lower dimensional manifold.

The authors have introduced AE models in [28] for approximating high dimensional nonlinear systems which arise when studying motion of insect or animal systems. We briefly summarize the theory in this section and explain how it is used in this paper. From any of a large selection of standard texts [96] on analytical mechanics or multi body dynamics, it is well-known that the governing equations for a system of rigid or elastic bodies that are connected by ideal joints can be generated from Lagranges equations with multipliers,

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_k}\right) - \frac{\partial T}{\partial q_k} + \frac{\partial U}{\partial q_k} + \frac{\partial \phi_i}{\partial q_k}\lambda_i = Q_k. \tag{3.1}$$

In Equation 3.1, $\{q_k\}_{k=1,\ldots,d}$ is a set of redundant, generalized coordinates for the mechanical system that is subject to m holonomic constraints $\phi_i(q(t)) = 0, i = 1 \ldots m$. Also, $T = T(q, \dot{q}, t)$ is the kinetic energy, $U = U_g(q,t) + U_s(q,t)$ is the potential energy which is the sum of gravitational and strain potential energies, λ is an m-vector of Lagrange multipliers, and Q is a d-vector of generalized forces. When the constraints $\phi_i(q) = 0$ are known explicitly in analytic form, it is a standard result of analytical mechanics that, at least in principle, it is possible to rewrite the system in terms of $d^* = d - m$ independent generalizes coordinates $q_i^*, i = 1, \ldots, d^*$ which eliminates the Lagrange multipliers. Of course, there are many mechanical systems that do not admit globally defined generalized coordinates and Equation 3.1 above can be extended by geometric mechanics to these systems by considering evolutions on a smooth configuration manifold Q_o [1, 19, 22].

The rigor afforded by techniques of geometric mechanics has an important implication for this paper: Lagrangian formulations can be introduced which are guaranteed under precise conditions to converge to solutions of the system with holonomic constraints and Lagrange multipliers [20]. In these approaches the approximations are solutions of the penalty parameterized unconstrained Lagrangian equations

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_k}\right) - \frac{\partial T}{\partial q_k} + \frac{\partial U}{\partial q_k} + \frac{\partial V_\epsilon}{\partial q_k} = Q_k \qquad k = 1, \ldots, d. \tag{3.2}$$

In Equation 3.2, $V_\epsilon := V_\epsilon(q;\epsilon)$ is an empirical potential energy function that has the property

$$\left[q : V_\epsilon(q;\epsilon) = 0\right] = \left[q : \phi_i(q) = 0, i = 1, \ldots, m\right]. \tag{3.3}$$

The reader is directed to [20] for a rigorous discussion of the convergence of trajectories that solve Equation 3.2 to motions $t \mapsto q(t)$ which also solve Equation 3.1. Reference [26, 28] derives probabilistic error rates of convergence for some such penalty formulations.

The important distinction between the two formulations is that Equation 3.1 is defined in terms of the holonomic constraints $\phi : \mathbb{R}^d \to \mathbb{R}^m$ and Equation 3.2 is defined in terms of an

energy function $V_\epsilon : \mathbb{R}^d \to \mathbb{R}$. When we say in this paper that the motion of an animal evolves on a submanifold contained within the high-dimensional configuration space, we do not in general know the associated constraints ϕ that would generate such a motion. Identifying ϕ directly would be an onerous task in all but the simplest cases because the dimension, m, is unknown and the function itself is nonlinear. Moreover, the number of dependencies, m, can be quite large compared to d. For human motion studies it is typical that $d = O(100)$ and the number of independent parameters $d^* = d - m$ can range from 1 to 3 [108, 109]. It follows that m is only slightly smaller than d. On the other hand, the penalty formulation in Equation 3.2 is written in terms of a single scalar valued function $V_\epsilon : \mathbb{R}^d \to \mathbb{R}$ which forces system trajectories to lie on the manifold without explicitly enforcing constraints. We will seek to exploit a special case of this strategy for the design of bioinspired joint models from observations of complex biomotions. Specifically, we will identify the empirical potential V_ϵ in the model above and use it to design the physical structure of a biomimetic bat elbow joint.

To design the physical joint structure, we use Equation 3.2 in two ways: once for the biological system under investigation, and again for our realization of the bioinspired hardware. For the biological system, we neglect the flexibility and gravitational effects of the constituent bodies of the bioskeleton and only seek to understand the structure of the submanifold that is associated with the large motions across the joint. Thus, both the gravitational and strain potential energies are zero and the identified empirical potential term V_ϵ pushes the system trajectory towards the identified manifold. Conversely, for the mechanical system V_ϵ is zero and the physical system potential U is non-zero. Because we are isolating the motion of a single joint, we further assume that the gravitational potential, U_g, is zero. Thus, the strain energy, U_s, is entirety of the potential stored in the joint. We subsequently optimize joint parameters so that the strain energy approximates the empirical potential as closely as possible. Furthermore, since this is our first experimental validation of the modeling technique described in [28], we only consider the static terms and the two models based on Equation 3.2 reduce to

$$\frac{\partial V_\epsilon}{\partial q_k} = Q_k \qquad \frac{\partial U_s}{\partial q_k} = Q_k. \tag{3.4}$$

In summary, we seek to identify the AE model, V_ϵ, and then approximate it using the physical strain energy of the joint, U_s.

Identifying the Analytical-Empirical Potential

The empirical potential V_ϵ in the above governing equations is in general a nonlinear function of the configuration variables q and is determined from experimental observations. The theoretical background for the pragmatic algorithm discussed in this section is rather abstract, especially during an initial exposure to the approach. Here, we review only the relevant theory needed to understand the approach in this paper. The reader is referred to the very brief overview of the theory in Appendix 3.6, the papers [31, 88] on learning theory, or our

discussions on applications to Lagrangian mechanics [28] for additional details. We assume that observations of the configuration

$$\{\overline{q}_i\}_{1\leq i\leq n} := \{\overline{q}_1, \ldots, \overline{q}_n\} \tag{3.5}$$

are available from experiments. The collection $\{\overline{q}_i\}_{1\leq i\leq n}$ are assumed to be random samples distributed according to an unknown probability measure μ on $Q_o \in \mathbb{Q}$. If we assume there are m unknown constraints ϕ that define the motion submanifold, Q_o, a critical defining property is that the potential V_ϵ satisfies Equation 4.15 making it an ideal substitute for the holonomic constraints. It is essential then, to choose the empirical potential such that $V_\epsilon(q) = 0$, at least approximately, on the motion submanifold. An elegant approach to this problem is achieved using the theory of reproducing kernel Hilbert spaces (RKHS), briefly summarized in the Appendix.

In principle, the construction of a RKHS of functions over the set $\mathbb{Q}$ as outlined in Appendix 3.6 enables the definition of an "ideal" potential function $V_\mu : \mathbb{Q} \to [0, 1]$ that has the property that

$$V_\mu(q) = 0 \qquad \Leftrightarrow \qquad q \in \text{support}(\mu) = \mathbb{Q}_0,$$

where we think of $\mathbb{Q}_0$ as the set of feasible states and μ as a probability measure which describes the distribution of states over the configuration space. Intuitively, the function V_μ takes a minimum value where the measure μ is concentrated (i.e. $q \in Q_o$), while it is large elsewhere (i.e. $q \notin Q_o$). When we define the scaled, ideal potential $V_{\mu,\epsilon}(q) := \frac{1}{\epsilon^2} V_\mu(q)$, we see that $V_{\mu,\epsilon} : \mathbb{Q} \to [0, 1/\epsilon^2]$ and the potential $V_{\mu,\epsilon} \to \infty$ as $\epsilon \to 0$ over regions outside of Q_0. If we could construct the ideal potential $V_{\mu,\epsilon}$, including it in a Lagrangian formulation of the dynamics would have the effect of driving trajectories toward the feasible region. Unfortunately, the ideal potential V_μ cannot be calculated in practice: it depends on an infinite dimensional operator that depends on the unknown measure μ.

In this paper we use the data-driven approximation V_ϵ of $V_{\mu,\epsilon}$ given by

$$V_\epsilon(q) = \frac{1}{\epsilon^2} V_{\overline{q}}(q) = \frac{1}{\epsilon^2}\left(1 - \frac{1}{n} K_{\overline{q}}(q)^T g_\lambda(\mathbb{K}) K_{\overline{q}}(q)\right) \tag{3.6}$$

where n is the number of observations or samples, $K_{\overline{q}}(q) := \{K(\overline{q}_1, q), \ldots, K(\overline{q}_n, q)\}^T \in \mathbb{R}^n$, $\mathbb{K}_{st} := K(\overline{q}_s, \overline{q}_t) \in \mathbb{R}^{n\times n}$ for $1 \leq s, t \leq n$, and $g_\lambda(\cdot)$ is a filter function as defined in [31]. In practice we compute

$$g_\lambda(\mathbb{K}) := U \text{diag}(g(\lambda_i)) U^T$$

with $\mathbb{K} := U\lambda_i U^T$ the spectral decomposition of the symmetric, real matrix $\mathbb{K}$. The filter function has a number of forms. For example, it can correspond to performing Tikhonov regularization or spectral cut-off over the eigenspaces. [88]

There are many practical reasons why the family of functions V_ϵ are well-suited to our overall strategy to construct robotic systems which exhibit desired motions that emulate animal motion. The empirical potentials V_ϵ are expressed in closed form in terms of the

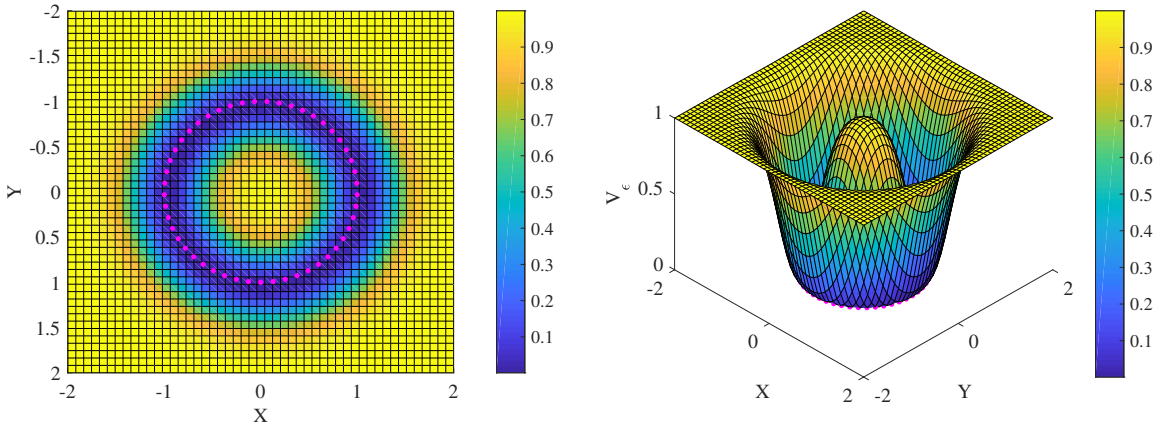
single kernel function $K(\cdot,\cdot) : \mathbb{Q} \times \mathbb{Q} \to \mathbb{R}$. It is easy to choose the kernel, $K(\cdot,\cdot)$ so that the functions V_ϵ are smooth, i.e., $V_\epsilon \in C^\infty(\mathbb{Q}, \mathbb{R})$. In this paper we choose the kernel function to be

$$K(\overline{q}, q) = \exp\left(-\beta||\overline{q} - q||^2\right), \tag{3.7}$$

where $\overline{q}$ is the basis function center, β is a hyper-parameter which controls the sharpness of K, and q is the location in the full configuration space at which the function is evaluated. This choice of kernel function ensures that V_ϵ is suitable for geometric mechanics formulations of Lagrangian mechanics on $\mathbb{R}^n$ and submanifolds of $\mathbb{R}^n$. Finally, since $V_{\overline{q}}(q) \in [0, 1]\ \forall q \in Q$, we know that $V_\epsilon(q) \in [0, \frac{1}{\epsilon^2}]\ \forall q \in \mathbb{Q}$. This last property gives a simple intuitive interpretation of the penalized empirical potentials. The set

$$Q_{\epsilon,o} := \left[q \in \mathbb{Q} \mid V_\epsilon(q) \leq 1\right] = \left[q \in \mathbb{Q} \mid V_{\overline{q}}(q) \leq \epsilon^2\right] \tag{3.8}$$

is a good approximation of the set in Equation 3.33 with Q_o the set of states that can be observed in principle. It is important to remember that the family of functions $V_{\epsilon,q}$ depend only on the samples $\{\overline{q}_k\}_{\{k=1,\ldots,n\}}$ and penalty parameter ϵ, and reflect only the geometry of approximations to the underlying set Q_o. The empirical potential in no way encodes information about the actual potential energy stored in a mechanical system.



(a) Top View (b) Isometric View

Figure 3.3: Empirical Potential Constructed Over Samples on the Unit Circle. Configurations associated with low potential energy (blue) occur near the sampled data (magenta dots) while configurations associated with high potential energy (yellow) occur away from the sampled data.

To illustrate these concepts more clearly, consider the following example. Suppose a particle moves around the unit circle. To construct the empirical potential energy function, we start by collecting a finite number of position measurements. For simplicity, we assume that all the samples are spaced evenly over the circle. Thus, the measurement set in Equation 3.5

contains entries $\overline{q}_i = \{\overline{x}_i, \overline{y}_i\}^T$. Using these samples and the kernel function in Equation 3.7, Equation 3.6 is evaluated over a uniform grid of points $q_i = \{x_j, y_j\}^T \in [-2, 2] \times [-2, 2]$. Figure 3.3 shows the empirical potential energy function over this interval.

In Figure 3.3, the empirical potential function takes a minimal value (dark blue) near the measurement set (magenta points) and a high value (yellow) everywhere else. Thus, if it is included in a Lagrangian model of the dynamics of the system, the empirical potential will push system trajectories towards the observed set. In other words, the dynamics of the identified system will evolve on $Q_{\epsilon,o}$.

In this paper, we present a methodology which uses constructions similar to this example to automate the design of robotic hardware which prefers configurations on the manifold identified from experimental observations of the system of interest—in this case, a bat in flight. The next step in the process is to approximate this empirical potential with a physically realizable potential U.

Approximating the Analytical-Empirical Potential

The actual, physical potential energy U can be derived for many mechanical systems from first principles. Such a function U will ordinarily depend on physical parameters p such as material and structural constants and we write

$$U(q) = U(q; p). \tag{3.9}$$

It will often be useful in discussing our method to normalize the empirical potential V_ϵ so that it is scaled to some reference value that makes physical sense. Suppose we have a measurement or accurate estimate of the potential energy of a physical system $U_r(q_r, p)$. We set $V_{r,\overline{q}}(q) := U_r(q_r, p)V_{\overline{q}}(q)$. Our general approach seeks to find the material and structural parameters in p for a candidate robot component or system design such that the actual potential energy matches the empirical energy $V_{r,\overline{q}}$ that only represents an ideal geometry,

$$U(q) \approx V_{r,\overline{q}}(q).$$
$$\text{Physical Energy} \approx \text{Empirical Energy}$$

In general, the parameters, p, are identified from a regularized optimization problem [84] of the form

$$p^* = \arg\min_{p \in P} ||V_{r,\overline{q}}(\cdot) - U(\cdot, p)||^2_{H_1} + \lambda ||V_{r,\overline{q}}(\cdot) - U(\cdot, p)||^2_{H_2} \tag{3.10}$$

with H_1 and H_2 function spaces with norms $||\cdot||_{H_1}$ and $||\cdot||_{H_2}$, respectively. In this expression, λ is the regularization parameter and H_2 is referred to as a smoothness space. Thus, increasing λ results in smoother solutions as measured in the H_2 norm. In this paper, the design of a robotic component or system is based on the solution of the regularized constrained optimization problem

$$p^* = \arg\min_{p \in P} \sum_{i=0}^{k} |V_{r,\overline{q}}(\overline{\xi}_i) - U(\overline{\xi}_i; p)|^2. \tag{3.11}$$

where $\left[\bar{\xi}\right]_{i=1,\ldots,k}$ a collection of collocation points which are used to match the energy functions as closely as possible.

With the prerequisite theory for learning empirical potential functions covered, we proceed in the next section by presenting the details of our analytical framework for designing optimal joint geometry from empirical observations of motion.

3.2 The Methodology

In this paper, we will use observations of the wrist location relative to the humerus fixed frame to optimize geometry which approximates motion of a bat elbow during straight and level flight. To simplify optimization, fabrication and experimental validation of the joint we project the experimental data onto SO(2) and assume a single rotational degree of freedom. The original data and its projection are shown in Figure 3.4. We assemble these measurements into the set $\bar{q} := \{\bar{q}_i \in Q_o \in \mathbb{R} \mid i \in \{1, \ldots, I\}\}$. In this paper we specifically choose a kernel function $K : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ to be the exponential function in Equation 3.7. With the observation set and the kernel function defined, the empirical potential function is calculated using Equation 3.6. A simple example which provides a physical interpretation of the purpose of this potential function is a spring which opposes motion away from the manifold. In synthetic experiments, we can use the identified $V_{\bar{q}}(q)$ directly; however, building a nonlinear spring which can approximately realize this energy function may be difficult or intractable. Because linear springs are much easier to fabricate, we formulate an optimization problem which assumes that the empirical potential function can be realized by a collection of linear beam elements that progressively load as joint angles near the boundary of the sample set are subtended. The kinematic definition for the joint architecture is presented in the following section.

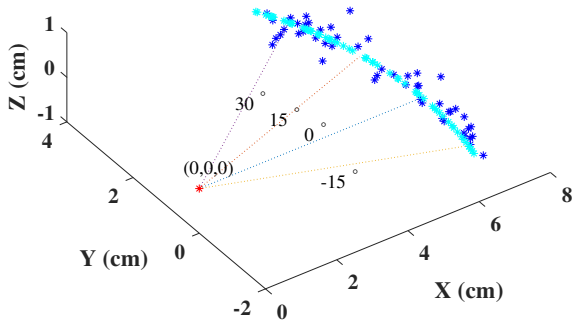


Figure 3.4: Wrist motion as observed from the Humerus frame. Dark blue points are the original data, light blue points are the data projected onto SO(2) which is the Lie group of proper rotations that, roughly speaking, are parameterized by one angle. The data on SO(2) is used to identify joint angles which are used to learn the empirical potential.

3.2.1 The System Model

In this section we introduce the system model and kinematics for a specific embodiment of a compliant joint. The joint presented in this paper is a single degree of freedom (DOF) pin joint with linear beam springs which effectively limit the range of motion of the articulated mechanism. The general system architecture is illustrated in Figure 3.5. Refer to the actual physical system in Figure 3.9.

In Figure 3.5 the articulated link, which approximates the radius of the bat arm, has length l_0 and rotates through angle q relative a rigid support which approximates the humerus of the bat arm. The compliant beam elements are shown below the articulated link and are anchored on the left to the humerus. As the articulated link is deflected, the compliant beams will store strain energy. This strain energy field is optimized to approximate the empirical potential energy which is identified using methods above.

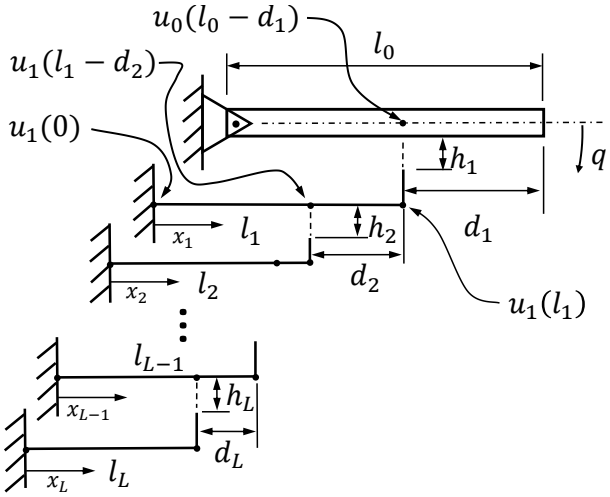


Figure 3.5: Compliant Joint Design. Link 0 is the articulated link and links 1 to L are the beam elements which approximate the empirical potential function.

While the kinematic model of the joint is shown to be nonlinear in the analysis that follows, we assume that the constituent beams depicted in Figure 3.5 deform such that a linear approximation is accurate. The transverse static displacement of each beam $i = 1, ..., L$ is assumed to satisfy the linear partial differential equation

$$\frac{\partial^2}{\partial x_i^2}\left(E_i(x_i)I_i(x_i)\frac{\partial^2 u_i}{\partial x_i^2}(x_i)\right) = 0 \tag{3.12}$$

where $u_i(x_i)$ is the vertical displacement of beam i at any point x_i along its length l_i. Assuming that the modulus of elasticity $E_i(x_i)$ and moment of inertia $I_i(x_i)$ are constant, and that the beams satisfy cantilever boundary conditions, a closed form solution for this

equation is derived in many standard texts [26, 51]. These references also derive the strain energy within each beam to be

$$U_i = \frac{1}{2}\int E_i(x)I_i(x)\left(\frac{\partial^2 u_i}{\partial x_i^2}(x_i)\right)^2 dx_i. \tag{3.13}$$

The well known finite element approximation of the strain energy is then factored into the form

$$U_i = \frac{1}{2}\boldsymbol{u}_i^T \mathbf{K}_i \boldsymbol{u}_i, \tag{3.14}$$

where $\boldsymbol{u}_i = \left[u_{i,1}, u_{i,2}...u_{i,2n_i}\right]^T$ is the vector of nodal displacements, n_i is the number of nodes in the i^{th} beam, and $\mathbf{K}_i$ is the stiffness matrix for beam i [26, 51]. Thus, to determine the energy stored in the beams, we need an expression for the nodal displacements of each beam. The nodal displacements $u_{i,j}$ of beam i are in fact nonlinear functions of the rotation q as discussed in detail in the next section.

3.2.2 Joint Kinematics

To derive an expression for the displacement of each compliant beam, we begin by defining the kinematics of our beam system. The displacement of each beam is determined by the configuration of the rigid link l_0. When a load is applied to the end of the rigid link of length l_0, it will deflect by an angle q such that the virtual work done on the system is equal to the energy stored within the beams. To compute the energy stored in the beams, we first determine the deflections of each beam. As the rigid link moves through angle q, the vertical deflection at the contact location of beam 1 is

$$u_0(l_0 - d_1) = (l_0 - d_1)\tan(q). \tag{3.15}$$

If the vertical deflection of the rigid link at the contact location of beam 1 exceeds the offset h_1, that is, $u_0(l_0 - d_1) > h_1$, then

$$u_1(l_1) = u_0(l_0 - d_1) - h_1. \tag{3.16}$$

Similarly, when displacement of $u_1(l_1 - d_2) > h_2$, it follows that the vertical displacement at the contact point of beam 2 is given by

$$u_2(l_2) = u_1(l_1 - d_2) - h_2. \tag{3.17}$$

Thus, the vertical displacement at the end of beam i can be expressed as a piecewise function of the displacement of beam $i-1$. We see that Equations 3.15 - Equation 3.17 are special cases of the recursion

$$u_i(l_i) = \begin{cases} 0, & u_{i-1}(l_{i-1} - d_i) < h_i \\ u_{i-1}(l_{i-1} - d_i) - h_i, & \text{otherwise.} \end{cases} \tag{3.18}$$

Using Equation 9, the tip displacement of any beam in a system of L beams can be determined in closed form. With tip displacements known, we derive analytical and numerical solutions for the displacements at any location on the beam having length l_i in the next section.

3.2.3 Recursive Formulation of Nonlinear Potential Energy

In this section, we derive an analytical solution for computing the potential energy stored within the beams as the robotic joint undergoes large rotations q. We assume that each beam is modeled with two or three nodes: one at the clamped end, one at the free end, and an optional node located a distance $l_i - d_{i+1}$ from the clamped end of beam i. The optional node is only included when beam i contacts beam $i+1$. Using the kinematics defined in Section 3.2.2, we develop an analytical algorithm for computing the loads, and subsequently the displacements, within each beam. The loading condition when the articulated link first contacts beam 1 is illustrated in Figure 3.6.

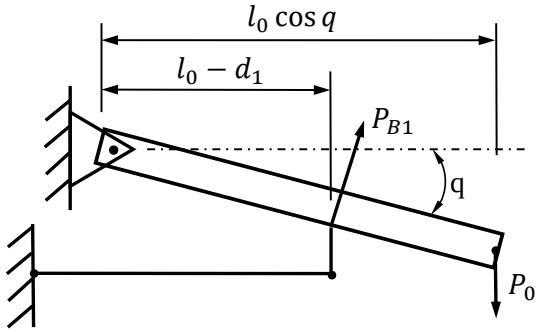


Figure 3.6: Loading Condition for Beam 1. when the articulated link contacts beam 1, the load applied to beam 1, P_{B1} acts perpendicular to the articulated link.

In the figure, a load, P_0 is applied to the articulated link. This load is resisted by the load P_{B1}. We assume frictionless contact such that only a transverse load is transfered to beam 1. After calculating the load applied to beam 1, the kinematics from Equation 3.18 are used to determine $u_1(l_1)$. Then we determine the angular displacement of the end of beam 1 using Castigliano's theorem.

$$P_1 = \frac{3E_1I_1}{l_1^3}u_1(l_1) = \frac{2E_1I_1}{l_1^2}\frac{\partial u_1}{\partial x_1}\bigg|_{l_1}, \tag{3.19}$$

$$\frac{3u_1(l_1)}{2l_1} = \frac{\partial u_1}{\partial x_1}\bigg|_{l_1}. \tag{3.20}$$

This relationship holds until beam 1 contacts beam 2. Then, the loading condition changes, and Equations 3.19 and 3.20 are no longer valid.

To determine if beam 1 has contacted beam 2, we solve for the deflection at the contact location for beam 2 and use the kinematics condition in Equation 3.18, where $i = 2$. If Equation 3.18 results in a non-zero displacement for the tip of beam 2, then the internal forces can be computed using a system of equations which is derived using the kinematics and Castigliano's theorem. We have the constraint

$$u_1(a_1) = u_2(l_2) \tag{3.21}$$

where $a_1 = l_1 - d_2$ and the load-displacement equation

$$\frac{P_1 a_1^2(3l_1 - a_1)}{6E_1I_1} - \frac{P_2 a_1^3}{3E_1I_1} = \frac{P_2 l_2^3}{3E_2I_2} + h_2. \tag{3.22}$$

Additionally, we know that

$$u_1(l_1) = (l_0 - d_1)\tan(q) - h_1, \tag{3.23}$$

$$\frac{P_1 l_1^3}{3E_1I_1} - \frac{P_2 a_1^2(3l_1 - a_2)}{6E_2I_2} = (l_0 - d_1)\tan(q) - h_1. \tag{3.24}$$

with $a_2 = l_2 - d_3$. Using Equations 3.22 and 3.24, we solve for the loads P_1 and P_2, and subsequently solve for the nodal displacements. This process is repeated in a similar fashion once beam 2 contacts beam 3.

More generally, we can compute the internal loads in a system of L beams by using the kinematics developed in Section 3.2.2 and Castigliano's Theorem. The general system of equations is

$$\frac{P_1 l_1^3}{3E_1I_1} - \frac{P_2 a_1^2(3l_1 - a_2)}{6E_2I_2} = (l_0 - d_1)\tan(q) - h_1, \tag{3.25}$$

$$\frac{P_i a_i^2(3l_i - a_i)}{6E_iI_i} - \frac{P_{i+1} a_i^3}{3E_iI_i} = \frac{P_{i+1} a_{i+1}^2(3l_{i+1} - a_{i+1})}{6E_{i+1}I_{i+1}} - \frac{P_{i+2} a_{i+1}^3}{3E_{i+1}I_{i+1}} - h_{i+1}, \tag{3.26}$$

$$\frac{P_L l_L^3}{3E_LI_L} - h_L = \frac{P_{L-1} a_{L-1}^2(3l_{L-1} - a_{L-1})}{6E_{L-1}I_{L-1}} - \frac{P_L a_{L-1}^3}{3E_{L-1}I_{L-1}}. \tag{3.27}$$

In the equations above $a_i = l_i - d_{i+1}$. In this paper we present two and tree beam implementations of the compliant joint. For the two beam system, only Equations 3.25 and 3.27 are required. For the three beam implementation, all three equations are used with $i = 2$ and $L = 3$. Proof of this is trivial and left to the reader. After these loads are computed, the deflections can be computed. Then a nonlinear function of the generalized coordinate q representing rotation of the rigid bar. The energy is computed as

$$U_K(q) = \begin{cases} \frac{1}{2}\boldsymbol{u}_1^T\mathbf{K}_1\boldsymbol{u}_1 & u_0 \geq h_1,\ u_{1,p} \leq h_2, \\ \frac{1}{2}\boldsymbol{u}_1^T\mathbf{K}_1\boldsymbol{u}_1 + \frac{1}{2}\boldsymbol{u}_2^T\mathbf{K}_2\boldsymbol{u}_2 & u_{1,p} \geq h_2,\ u_{2,p} \leq h_3, \\ \frac{1}{2}\boldsymbol{u}_1^T\mathbf{K}_1\boldsymbol{u}_1 + \frac{1}{2}\boldsymbol{u}_2^T\mathbf{K}_2\boldsymbol{u}_2 + \frac{1}{2}\boldsymbol{u}_3^T\mathbf{K}_3\boldsymbol{u}_3 & u_{2,p} \geq h_3. \end{cases} \tag{3.28}$$

Note that this is a minor abuse of notation due to the stiffness matrices having different numbers of rows and columns, but it is apparent from context that the $\mathbf{K}$ with the largest subscript only has two rows and two columns. While equation 3.28 is developed for three beams, it can be extended to L elements trivially. Equation 3.28 is a piecewise quadratic energy function which is optimized to approximate the empirical potential function in Equation 3.6. Details of the optimization for our experiments will be discussed in 3.3.

3.3 Numerical Optimization

The empirical energy function $V_{r,\bar{q}}(q)$ in Equation 3.6 is a summation of exponentials and the hardware realization developed in this paper uses piecewise quadratic springs. Therefore, we choose to design the joint by solving the optimization problem

$$p^* = \arg\min_{p \in P} \sum_i |V_{r,\bar{q}}(\bar{\xi}_i) - U_K(\bar{\xi}_i; p)|^2, \tag{3.29}$$

where i indexes over a set of quadrature points chosen by the designer and P is a set of admissible parameters. The admissible parameters P in Equation 3.29 can include any desired beam parameters, namely E_i, I_i, l_i, h_i, and/or d_i. The optimization should be constrained such that all these terms are positive because they are physical properties of the system. While this constraint does reduce the size of our solution space, there are many other practical constraints which can be imposed to restrict our solution space even further.

One important constraint is to limit beam parameters such that the beams are of approximately the same stiffness. One optimization condition which enforces this is selected to be

$$I_1 \leq 2I_2 \leq 2I_3. \tag{3.30}$$

This constraint ensures that all moments of inertia are within 100% of each other. Additionally, we place an upper limit on the sum of offsets, h. This restricts the size of the mechanism and prevents small energies from dominating the optimization. Similar constraints can be added to any of the beam parameters. For the experiments in the paper we only optimize over the moments of inertia and offsets. Optimizing over moments of inertia is beneficial because the energy function scales linearly with this parameter. Optimizing over h changes the locations of the discontinuities to improve the fitness of the realized energy function.

3.4 Results

In this section, we validate our theoretical and analytical framework for optimizing compliant structures around conventional pin joints. To this end, we fabricate three examples

of compliant joints: two from synthetic data and one from experimental data. All three joints were designed in Autodesk™ Inventor™ and printed using a Monoprice Maker Select V2 3D printer. The printer fabricates parts from polylactic acid (PLA) which has a wide range of reported elastic moduli. Furthermore, printing parameters such as percent infill, infill geometry, layer thickness, and print speed can have adverse effects on the print quality and final strength of the material. Thus, modulus of elasticity is determined by performing bending tests on cantilever beams which are printed using the same parameters which will be used to print the optimized joints. After determining the modulus of elasticity, we present validation experiments for the optimized joints.

To determine the modulus of elasticity for our PLA, we clamp 2 inch long 3D printed beams in a vice, and measure their load-deflection curves. Each beam is loaded using a Dynamixel RX-24F servo and pulley to which a cable is attached. The cable connects to a a Transducer Techniques TBS-40 Full-Bridge Thin Beam Force Sensor with associated mounting hardware which are used to measure the load applied to the beam. The other end of the force sensor is attached to the end of the beam with a second length of cable. The beam is loaded by commanding the servo to a new angle displacement. Commands are sent to the servo by using the Dyanmixel Wizard and associated USB to TTL converter provided by the manufacturer. To measure the force applied at a particular displacement, the force sensor is connected to a Keyes234 breakout board for an HX711 full-bridge amplifier which filters the signal generated by the force sensor and performs analog to digital conversion of the signal. The serial output lines on the breakout board are connected to an Arduino Uno which runs the Sparkfun HX711 Arduino library and transmits the force measurements via USB to the computer. Finally, the deflection at the tip of each beam is measured using a Mitutoyo height gauge with a resolution of 0.001 inches. To insure result consistency with respect to beam thickness, bending tests are conducted with beam thicknesses of 0.068, 0.085, 0.104, 0.124, and 0.144. All beams are fabricated with a width of 0.202in. The test system is shown in Figure 3.7.

Using this system, we measure the load at various tip deflections. Figure 3.8 shows the force-displacement relationship for all of the beams tested. The figure shows that all thickness tested have a strongly linear force-displacement curve. Thus, Castigliano's theorem can be applied to determine the modulus of elasticity of the PLA used in these experiments. Ideally, the modulus calculated for each of the 5 beams should be identical. The computed modus of elasticity of was 308 ksi with a standard deviation of 21 ksi.

With the modulus of elasticity determined, the optimization problem in Section 3.3 was completed to design three different bioinspired joints. First, synthetic data was created by assuming an angular deflection of ±15deg. An empirical potential was fit to samples that were evenly distributed through

the range of motion. During optimization, the moment of inertia, I_i, and offset, h_i for each beam was allowed to vary. The upper limit for the sum of the offsets h_i is 0.158in. One complaint joint was designed using 2 beam elements and another was designed assuming 3

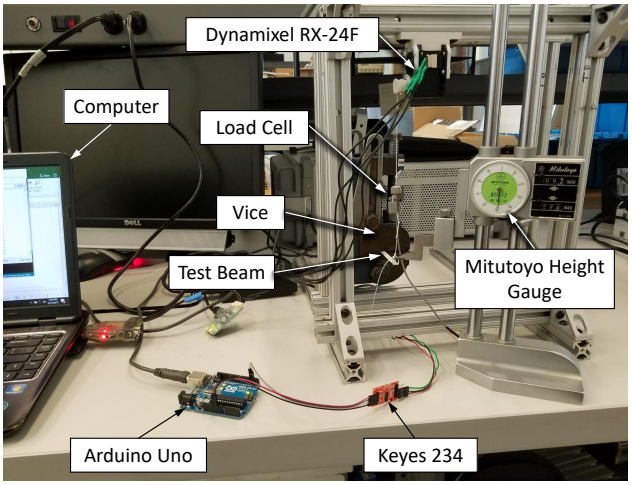


Figure 3.7: Modulus of Elasticity Test. The computer controls the Dynamixel. The Arduino and Keyes 234 are reads the load cell. The height gauge measures the beam deflection.

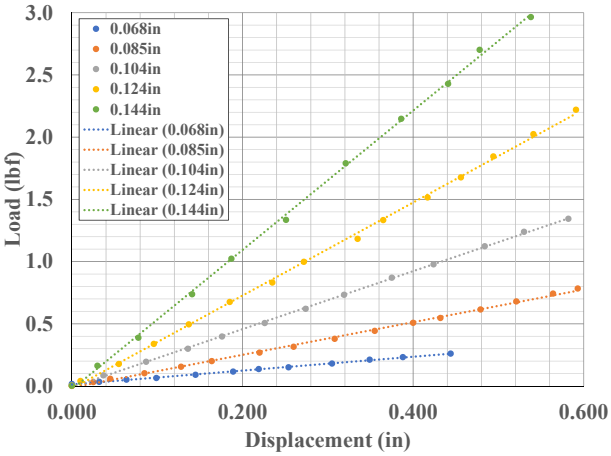


Figure 3.8: Force Displacement Functions for Linear Beam Testing. Five different thickness beams were tested to generate load displacement functions. Beam thicknesses tested were: 0.068" (blue), 0.085" (red), 0.104" (gray), 0.124" (yellow), and 0.144" (green). As shown in the figure all of the force deflection curves are strongly linear. Therefore, we can assume linear beam bending through these ranges of deflection.

beam elements. In addition to the synthetic data, elbow angles of a Pratt's round leaf bat (H. pratti) during straight and level flight were used to design a third joint. This data was collected using the methods presented in [14]. The identified parameters for all three joints are provided in Table 3.1.

The three joints in Table 3.1 were printed using a Monoprice Maker Select V2 printer. Layer resolution was set to 0.2mm. The printed joints are shown in Figure 3.9. The joint on the left is generated using synthetic data and the thickness, t, and vertical offsets, h_1 and h_2,

of two beams are optimized. The joint in the middle is generated using synthetic data and thicknesses and offsets of three beams are optimized. The joint on the right is the 3 beam design generated using motion capture data of the bat elbow. This the compliant structure surrounding this joint is also contains three beam elements. The articulated link pinned to the joint on the right can be removed and placed on any of the other specimens during testing.

Parameter	2 Beam Synthetic	3 Beam Synthetic	3 Beam Experimental
I_1 [in^4]	1.25e-5	4.65e-6	1.83e-6
I_2 [in^4]	2.50e-5	1.86e-5	1.65e-5
I_3 [in^4]	N/A	9.30e-6	5.5e-6
h_1 [in]	0.118	0.075	0.118
h_2 [in]	0.038	0.050	0.059
h_3 [in]	N/A	0.033	0.020

Table 3.1: Joint Parameters Identified by Optimization

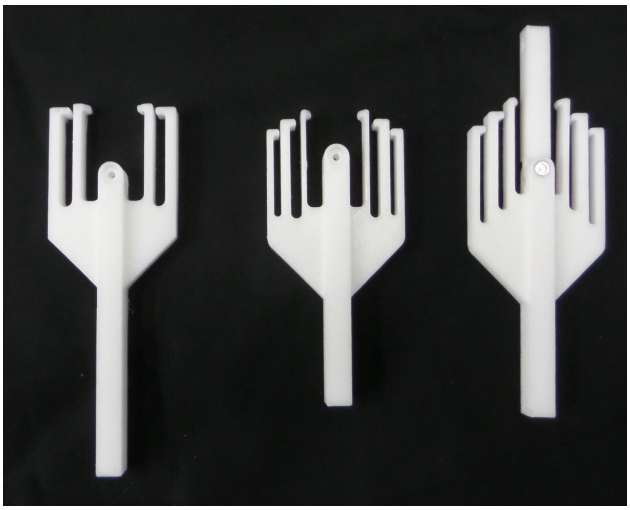
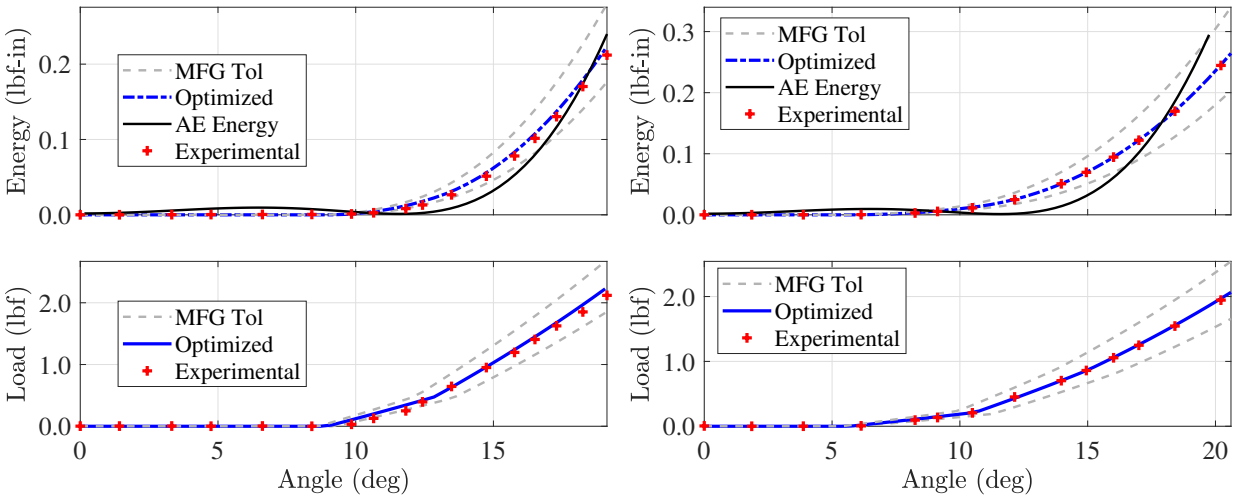


Figure 3.9: Three Compliant Articulated Mechanisms Designed from Synthetic or Experimental Data. Left: 2 beam specimen generated from synthetic data. Middle: 3 beam specimen generated from synthetic data. Right: 3 beam specimen generated from motion capture data of the elbow shown with articulated link attached. The articulated link is removable and can be installed on each of the test pieces. All parts were 3D printed using a Monoprice Maker Select V2 printer with a layer resolution of 0.2mm

To validate the energy functions for each of the optimized joints, the beam bending test system, shown in Figure 3.7, is used. As in the beam bending tests, the servo is used to apply force to the free end of the articulated mechanism, deflection at the free end of the mechanism is measured with the height gauge, and the applied force is measured with the load cell. The load cell and height gauge are zeroed when the articulated link and fixed link are parallel. The actuator progressively loads the beam and force-displacement data is recorded. The force-displacement and energy-displacement function for each compliant mechanism is shown in Figure 3.10.

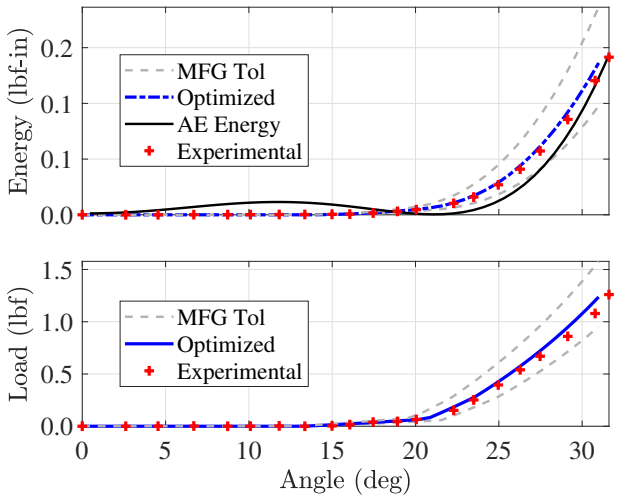
In Figures 3.10a through 3.10c, the bottom plots show the force-displacement measurements obtained using the load cell and height gauge. The blue line is the optimized force-deflection curve, the gray dashed lines denote force displacement curves for joints printed with $\pm 100\mu$m tolerances which is the maximum resolution of the printer used to fabricate the joints. Finally, the red '+' marks denote the experimental force-displacement measurements. Due to the offset of the first compliant beam, all of these plots show a "dead-zone" in which zero force is required to achieve a wide range of deflection. The ends of the dead-zones occur at 9^o, 6^o, and 13^o for the two-beam synthetic, three-beam synthetic, and three-beam experimental joints, respectively. Furthermore, these plots show bends in the force displacement curves at 13^o, 11^o, and 21^o, respectively. This behavior is observed due to the offset between the first and second compliant beams. Finally, the three beam joints show a final bend in the force-displacement curves at 15^o and 24^o for the three-beam synthetic and three-beam experimental mechanisms, respectively. These changes in slope are less noticeable because the optimized thicknesses of the third beams are much thinner than the optimized thicknesses of the second beams as reported in Table 3.1. The data shown in these force-displacement plots is used to compute the experimental energy-displacement curves used for validation of the energy optimization method presented in Section 3.2.1.

The energy-displacement curve for each compliant joint is shown in the top of Figures 3.10a through 3.10c. In the plots, the black line is the analytical-empirical potential function, the blue line is the optimized energy function and the red '+' marks are the experimental energy. The experimental energy is computed using a mid-point Riemannian sum over the experimental force-displacement data. The gray dashed lines represent confidence bounds which reflect minimum and maximum stiffnesses given manufacturing tolerances of $\pm 100\mu m$ on the optimized parameters. The optimized energy function approximates the analytical empirical energy function as closely as possible. The discrepancy between these curves occurs because the piecewise quadratic energy function cannot exactly match the sum-of-exponential form of the analytical empirical energy function. Furthermore, the experimental energy is close to the optimal energy and is within the bounds established from manufacturing tolerances. Thus, our analytical and theoretical framework for designing joint geometry is validated for the single degree of freedom static case.



(a) 2 Beam Synthetic Data

(b) 3 Beam Synthetic Data



(c) 3 Beam Experimental Data

Figure 3.10: Load-Deflection and Energy-Deflection functions for the compliant mechanism designed using synthetic data and three beam elements. The analytical-empirical energy function (black line) fits over the range of sample points. The optimized load and energy (blue) are close to the experimental load and energy (red with '+' markers). The experimental values are within the confidence bounds (gray dashed) which are computed assuming $\pm 100 \mu m$ tolerance on the optimized beam height and offsets.

3.5 Conclusion

In this paper present an analytical and theoretical framework for generating compliant structures around pinned joints by using empirical observations of bat flapping flight. The motion

capture data used for this work was collected in previous investigations [12, 14]. Using this data, we construct an analytical-empirical potential function over the observed configurations. We then optimize a compliant structure around the joint which prefers configurations on or near the submanifold identified from the observed configurations. Experimental validation of our algorithm is presented for both synthetic and experimental motion data. The experimental validations show force and energy functions which closely match the empirical beam energy identified using our framework. While the joints constructed for this paper are rather simple, they successfully demonstrate the methodology presented herein, and we will expand on the hardware in future investigations.

The ultimate goal of this research is to generate a general framework for automating the design and fabrication of robot joint geometry from empirical observations of biomotion. Thus, in the future we will solve the more general problem of generating geometry which prefers motion on an arbitrary manifold in $\mathbb{R}^N$.

3.6 Appendix A: Reproducing Kernel Hilbert Spaces (RKHS)

The empirical potential function V_ϵ is constructed in terms of a kernel function K that defines a reproducing kernel Hilbert space of functions over $\mathbb{Q}$. A real-valued function $K(\cdot,\cdot) : \mathbb{Q} \times \mathbb{Q} \to \mathbb{R}$ is referred to as a kernel function on $\mathbb{Q} \times \mathbb{Q}$ provided that it is symmetric and continuous. For each $\bar{q} \in \mathbb{Q}$ we define a basis function $K_{\bar{q}}(q) := K(q, \bar{q})$ in terms of the kernel K: this is known as the basis function centered at $\bar{q}$. In most practical problems, the kernel K looks like a "bump function" in that it decays very rapidly far away from its center $\bar{q}$. In fact, it often takes the form of a radial basis function which is written as $K(q, r) := \hat{K}(\|q - r\|)$ for some function $\hat{K} : [0, \infty) \to [0, \infty)$. There are a wide selection of such kernels that are commonly used for scattered data approximation including exponential functions, multiquadrics, and thin plate splines [117]. In our numerical studies we select the exponential kernel $K(x, y) := e^{-\beta\|x-y\|^2}$, for example, but any other of these popular kernels could also be used.

The RKHS H is defined to be the closed linear span of the the set of functions $\{K_q(\cdot)\}_{q\in\mathbb{Q}}$ [84]. In other words, the RKHS consists of all those functions $f : \mathbb{Q} \to \mathbb{R}$ such that

$$f(\cdot) = \lim_{k\to\infty} \sum_{i=1}^{k} \alpha_i K(\bar{\xi}_i, \cdot) \tag{3.31}$$

for sequences of coefficients $\{\alpha_i\}_{i=1\ldots k} \subset \mathbb{R}$ and locations $\{\bar{\xi}_i\}_{i=1\ldots k} \subset \mathbb{Q}$. The determining property that makes H a RKHS is that the kernel K satisfies the reproducing property $(f, K_q)_H = f(q)$ for all $q \in \mathbb{Q}$ and $f \in H$. When K is sufficiently regular, it is possible to interpret the RKHS space H as a subset of the μ-weighted Lebesgue space $L^2(\mathbb{Q}, \mu)$.

This inclusion is represented by an injective, continuous mapping $i_K : H \to L^2(\mu)$ with $i_K : f \in H \mapsto i(f) = f \in L^2(\mu)$. When the kernel K is also bounded, it can be shown that it, as well as its adjoint $i_K^* : L^2(\mathbb{Q}, \mu) \to H$, are compact. In this case the operators $T_\mu := i_K^* i_K : H \to H$ and $T_K := i_K i_K^* : L_\mu^2(\Omega) \to L_\mu^2(\Omega)$ are compact and self-adjoint. Our empirical potential energy functions are defined in terms of the operator T_μ. See [88] for a detailed discussion of the properties of these infinite dimensional operators, their eigenvalues, and their eigenfunctions.

We define the ideal potential energy, corresponding to the samples distributed by the unknown measure μ on $\mathbb{Q}$ as

$$V_\mu(q) := 1 - \left(T_\mu^+ T_\mu K_q, K_q\right)_H . \tag{3.32}$$

This ideal potential energy has the property that $V_\mu : Q \to [0, 1]$. If the distribution μ only generates samples in a subset $Q_o \subseteq \mathbb{Q}$, we have

$$\text{support}(\mu) = Q_o = \left[q \in \mathbb{Q} \quad | \quad V_\mu(q) = 0\right] . \tag{3.33}$$

That is, the support of the probability measure μ is Q_o [31, 88]. In other words, V_μ takes a minimum value over the set of ideal or possible observations and is non-zero otherwise. This means that the probability of generating samples outside of Q_o is zero. Roughly speaking, choosing a potential energy $V_\epsilon(q) := \frac{1}{\epsilon^2} V_\mu(q)$, has the effect of "pushing" system trajectories away from high potential values and onto the set of observations. This effect becomes more pronounced as the penalty parameter $\epsilon \to 0$. Unfortunately, we cannot use $V_\mu(q)$ in practice since it depends on the unknown probability measure μ that describes the distribution of measurements over $\mathbb{Q}$ and involves the generally infinite dimensional operator T_μ. Instead we construct a sampling approximation $V_{\overline{q}}$ defined as

$$V_\epsilon(q) := \frac{1}{\epsilon^2}\left(1 - \left(g_\lambda(T_{\overline{q}}) T_{\overline{q}} K_q, K_q\right)_H\right) \tag{3.34}$$

$$= \frac{1}{\epsilon^2}\left(1 - \frac{1}{n} K_{\overline{q}}(q)^T g_\lambda(\mathbb{K}) K_{\overline{q}}(q)\right) \tag{3.35}$$

where $K_{\overline{q}}(q) := \{K(\overline{q}_1, q), \ldots, K(\overline{q}_n, q)\}^T \in \mathbb{R}^n$, $\mathbb{K}_{st} := K(\overline{q}_s, \overline{q}_t) \in \mathbb{R}^{n \times n}$ for $1 \leq s, t \leq n$, $S_{\overline{q}}$ is the sampling operator $S_{\overline{q}} f := \{f(\overline{q}_1), \ldots, f(\overline{q}_n)\}^T \in \mathbb{R}^n$, $T_{\overline{q}} := \frac{1}{n} S_{\overline{q}} S_{\overline{q}}^*$, and $g_\lambda(\cdot)$ is a filter function as defined in [31]. Perhaps it is easiest to think of $g_\lambda(\cdot)$ as an approximate pseudoinverse that neglects extreme singular values. It is the expression $V_{\overline{q}}(q) := 1 - \frac{1}{n} K_{\overline{q}}(q)^T g_\lambda(\mathbb{K}) K_{\overline{q}}(q)$ that constitutes a practical algorithm for approximating the ideal potential $V_\mu(q)$.

Chapter 4

Learning Bioinspired Joint Geometry from Motion Capture Data of Bat Flight

Bioinspired robotic systems are often designed by assuming that the kinematics of the biological system of interest are well approximated by an open kinematic chain subject to box constraints. These box-constrains are typically generated by designer interpretation of biomotion studies or pragmatic fabrication constraints. In contrast to this standard design paradigm, this paper presents a methodology for learning joint geometry which restricts the range of motion of generic ball joints to only the reachable set observed during biomotion experiments. This reachable set is identified by constructing in analytical-empirical potential energy function over the experimental observations. The energy function "pushes" configurations close to the set of observations. This energy function is then thresholded to identify the zero-potential configuration set, and joint geometry is constructed using the resulting contour. We construct an entire bat wing using this method, and demonstrate through motion capture experiments that the learned geometry successfully restricts motions to the zero-potential set.

4.1 Introduction

In recent years, bioinspiration has become an ubiquitous theme in robotics due to potential improvements in the adaptability, maneuverability, and efficiency with which biological systems interact with their environments. To date, many researchers have studied the motion of humans [72], insects [99], birds [77], bats [102], fish [58], and many other biological systems as a basis for robotic design. Typically these studies record observations of different regimes of motion using high speed video equipment. The video sequences are then digitized to provide either an intuitive, qualitative understanding of animal motion or a quantitative analysis which includes two or three-dimensional reconstruction of the observed motion. This data is then used to design robotic systems which imitate their biological counterparts.

Most often, the first step in designing robotic systems is to approximate the biomotion observed during experiments by assuming that the skeletal mechanics occurs within the space

$SE(3)\times SO(n_1)\times SO(n_2)\times\ldots\times SO(n_l)$ where l is the number of links in the kinematic chain, $SE(3)$ accounts for the body translation and rotation, and $SO(n_i)$ is the special orthogonal group for a joint with n_i mutually orthogonal rotations. The motion of the system is typically restricted to a submanifold of this configuration space by enforcing box constraints on joint motion. In many studies, the box constraints are identified by a designer or are imposed by pragmatic considerations related to fabrication constraints. This simplification, however, is a departure from the actual motion observed during experiments. It should be noted that this design paradigm yields very capable and impressive robotic systems. However, it requires moderate to substantial intervention by an expert analyst, may be difficult to identify correctly, may lead to admissible configurations which are unnatural or unstable, and even the optimal solution may differ from the original kinematics in important ways. Thus, the goal of this paper is to establish a rigorous methodology for learning robotic joint geometry directly from biomotion data.

Our framework uses motion capture data of bats in flight and recent developments in learning theory to formulate an algorithm for automated processing of biological motion data into robotic joint geometry. We use experimental observations to learn the existence of a motion submanifold within the full configuration space by formulating an analytical-empirical potential energy field over the set of poses observed during experiments. This energy field "pushes" system trajectories towards those configurations. The energy field is used to identify the a zero-potential (ZP) set of admissible configurations. The boundary of the ZP set is used to design a range-of-motion limiting end cap which only permits motions on or near the identified submanifold.

To motivate this work we begin with a review of recent advancements in bioinspiration of robotic systems which use experimental observations of biological motion as a foundation for designing robotic systems. Then, we describe our methods for collecting biomotion data of bat flight. Next, we present the learning theory which identifies an empirical potential energy function over the biomotion data. Then, we discuss how the energy function is used to drive the design of joint geometry. Finally, we present results from biomotion experiments, energy function learning, hardware fabrication, and validation experiments which demonstrate that our method successfully designs range-of-motion limiting geometry directly from motion capture data. We conclude with a few closing remarks and present opportunities for future study.

4.2 Recent Developments in Bioinspired Robot Design

4.2.1 Walking Robots

Due to the wide range of potential applications, human motion studies are arguably the most prolific in the field of biological motion capture. Human pose estimation has been used

to drive realistic animations, design robotic hardware, improve prosthesis design, and study musculo-skeletal pathologies. A comprehensive review of such applications is presented in [47]. Due to the wide range of applications, many benchmark datasets have been created [30, 93]. In such datasets, many regimes of motion data are available for open-source download and inclusion in new research. These studies of motion have been used to design, fabricate, and control many different humanoid robots.

Humanoid robots are of substantial interest due to their ability to operate in similar environments as humans and leverage the use of tools designed for use by their human counterparts. To date, many impressive and highly-functional examples of such robots have been created such as Lucy [110], HUBO [78], HRP-3 [55], Athlete [76], and Petman [75]. In all of these studies, human kinematics are approximated by an open-kinematic chain of rigid links connected by ideal rotational joints. Each joint is subject to range of motion limitations known as box-constraints. While the constraints are identified from motion studies, they are either generated by designer interpretation of biological kinematics, or they are imposed by practical considerations made due to fabrication restrictions. Additionally, limited discussions of how the constraints are determined is given beyond stating the number of DOF per joint and the range of motion. These designs are excellent examples of bioinspired robots which use designer interpretation of motion capture data as a basis for design.

In addition to using motion studies for design, anatomical studies of humans are often used to propose alternative designs. For example, authors in [67] design a bipedal walking platform using a system of parallel actuators for the leg units. Orientation and position of each foot is controlled using 6 linear actuators for each leg. Actuator placement is loosely based on the human musculo-skeletal system. Another work, [70] presents the updated design of a musculo-skeletal humanoid robot which incorporates a shoulder-blade inspired shoulder joint which improves the range of motion of a traditional ball and socket joint employed to approximate the human shoulder. The authors note that the complex shoulder geometry made hand positioning difficult. This issue was overcome by using a cell phone camera sensor to track motion of the shoulder blade with respect to the chest. While these works give a more detailed discussion of kinematic design and mimic anatomical features of humans, they do no learn joint geometry directly from motion data.

In addition to the human motion studies, may animals have been studied to determine how morphological attributes affect their motion. In [29], the scaling of limbs and posture within the cat family (felidae) are investigated. The authors compare the morphology and stances of 9 different species of cats by recording video of the animals walking. In the study, planar kinematics of hindlimb and forelimb motion are analyzed. Additionally, [50] presents a detailed anatomical study of the hindlimb of a cheetah compared to that of a greyhound. Authors aim to determine morphological differences which enable the cheetah to run at 29 m/s while the greyhound only runs at 17 m/s. Surprisingly, the authors find that the hip extensor muscle in the cheetah has a smaller volume than in the greyhound. However, the authors posit that the cheetah runs faster due a greater amount of back activation and a larger hip stabilizer which is used for resisting pitching moments induced during high speed

running. These studies have been used as the basis of robotic system design.

For example, in [89] the authors develop a cat-robot with biomimetic hindlimbs supported by a rolling cart. This robot is used to study the role of muscular sensory signals at the limbs in terminating the stance phase of walking gaits. Research has shown that cats, while physically smaller and of smaller brain size, move faster and with more agility than dogs. This suggests that differences in locomotion arise from morphological considerations and not control or intelligence. Beyond citing [29, 50], minimal discussion of the kinematic design of the robot, beyond stating that the legs have 3 DOF each and 4 pairs of flexion-extension muscles, is presented. As in the design of humanoid robots, the authors of [4] supplement motion analysis with anatomically correct features to design a leg inspired by a cheetah for use in the MIT Cheetah Robot [91]. The inertia of the leg is reduced by mimicking the biological structure of bone using a foam-core, resin-shell, shape-deposition manufacturing method. Furthermore, internal stresses are reduced by adding a passive Achilles tendon to the lower limb. These robots are impressive, functional realizations of robotic hardware designed from expert interpretation of biological motion and morphology. Due to the success in designing walking robots using this paradigm, other investigations have used similar methods to design robotic manipulators.

4.2.2 Robotic Manipulators

In the design of robotic hands, both kinematic analysis of video and morphological studies of the musculo-skeletal system provide inspiration for high-dexterity robotic manipulators. In [57], the authors develop a novel experimental protocol to measure the force exerted by the human thumb and index finger. The experimental hardware accounts for soft tissue effects and slippage. Kinematic measurements are also recorded using a Vicon^{TM} motion capture system. Additionally, muscle activation is measured using an sEMG sensor. The authors use this data to design and control a two-finger manipulator. This study simplifies the kinematics of the hand and only considers planar kinematics of the thumb and index finger.

In addition to kinematics and force data driven design, a substantial amount of work has been done in using anatomical studies of the human hand as a basis for design. Authors in [27], use anatomically correct rigid links, elastic ligament connections, and antagonistic tendons to construct a robotic manipulator inspired by the human hand. The authors note that simplified, constant-DOF mechanisms which resemble the human hand have been created in the past but this restricts motion of the device. Their approach enables substantially larger forces to be applied to manipulated objects and the range of motion of the hand is dramatically increased. Their goal is to incorporate compliance in the hand while retaining forceful interaction with the environment and increasing the range of motion of the hands. However, fabrication of the mechanical structure is based on anatomy and not driven by experimental observations of motion.

4.2.3 Flapping Wing Robots

In addition to the study of human motion, the synthesis of flapping wing robots has gained renewed attention in recent years due to promises of increased performance based on bioinspiration. Typically these robots resemble that of an insect or bird [77]. Many designs include panel shaped or flexible plate wings which are actuated at the root. Recently, investigators have introduced passive degrees of freedom which decrease air resistance on the upstroke by enabling the wing surface to flex and extend during flight [18, 74]. In addition to flexion-extension compliance, authors in [118] add a compliant wing spar to mimic avian wrist actions allowing pronation-supination of the wing surface. The authors present experimental results which show that the optimized compliant mechanism increases the upward acceleration of the body just before the upstroke-downstroke transition. These works are excellent examples of adding an optimal, compliant structures to increase performance, but the designs are based on qualitative analysis of bird wing kinematics.

In addition to bird and insect robots, bat flight has been studied as inspiration for flapping wing robots. The seminal research presented in [15, 16, 24, 52, 53] studies the kinematics of bat flight in detail. In addition to determining the foundational kinematics, researchers in [87] perform principle orthogonal decomposition (POD) to determine which kinematic variables have the highest contribution to the overall motion of the bat. Such studies of motion are then used to create robotic hardware detailed in [25] which contain the most important degrees of freedom. This hardware is placed in a wind tunnel to measure lift and drag forces and compute C_l and C_d curves which are widely available for standard fixed wing MAVs but are extremely difficult to generate for flapping wing robots. This substantial body of research is then used to develop robots such as the one described in [81, 82, 83]. As in [25], the robot, Bat Bot B2, is designed using the POD analysis in [87] to determine the most important degrees of freedom and actuation. In contrast to the walking robots discussed above, this work presents a detailed description of how biomotion studies of bats in flight are used to inform kinematic robotic design. The reason for the care in specifying kinematics is due to weight restrictions of flying robots which are not as stringent for walking robots. However, the kinematics of the bat robot constructed still differ from the kinematics of the biology [45] and ultimately, a designer is responsible for specifying all aspects of wing design.

4.2.4 Trends In Bioinspired Robotic Design

In the body of research reviewed above, there exist three paradigms for kinematic design of bioinspired robotic hardware: 1) observe motion, construct a kinematic model, and identify box constraints on the assumed degrees of freedom, 2) study the anatomy in detail and model bones and joints based on morphological aspects of the subject, and 3) use a combination of methods 1 and 2. These paradigms have been used to create bioinspired robots with impressive functionality. However, many have noted that the configurations achievable by the robots deviate from the biological counterparts which they aim to imitate [45]. Many works

have focused on path planning or control algorithms for compensating for fundamentally flawed kinematics.

In [121], the authors propose an optimization scheme for matching the pose of a humanoid robot to that of a human actor performing Taiji. The algorithm substantially improved the robot's ability to mimic the recorded motions. The authors note that the difficulty in matching pose is caused by differences in kinematics. Box constraints on joint angles of the robot are one limitation which prevents the robot from exactly matching the pose of the human. Additionally in [60], the authors propose a method for using Kinect sensor measurements of human motion to drive the motion of a Nao robot. The authors propose clever methods for overcoming differences between the robot and the human which makes transitions between certain poses unstable. These investigations which attempt to correct flawed kinematics motivate the need for developing rigorous design methodologies which preserve the observed motion of the biosystem under study.

In contrast the methods presented above, our approach presents an alternative design paradigm: collect experimental observations of motion, select a general joint architecture, and use the motion data to directly drive the detailed design of the joint such that only motions observed during experiments are admissible. This methodology is similar to previous works in that we collect multiview video of the biological system we are studying and that we must select a general joint architecture. However, Our method is unique in that we use empirical observations of motion to identify an analytical-empirical potential energy field which pushes motions towards configurations observed during experiments. We then use this energy field to automate the design of geometry which restricts motions to those observed during experiments.

4.3 Experimental Methods

The joint geometry designed in this paper is learned from experimental observations of a bat performing straight and level flight. To collect experimental measurements, a low-cost, highly-redundant imaging system based on the GoPro Hero 3+ Black is used to collect video of the flight motion from 21 different viewing angles. While the redundant nature of the viewing system successfully minimizes occlusion, the consumer grade cameras introduce significant amounts of radial distortion in the images. The video is used to reconstruct the three-dimensional motion of over 200 marker points distributed across the bat's body and wings. To learn joint geometry, only the markers on the joints of the bat skeleton are used. Extensive discussions about distortion removal and reconstruction of the motion is reported in [12, 13, 14].

In [13, 14] the authors use the well-known Devanit-Hartenberg (DH) convention to assign rotational degrees of freedom to each of the joints on the wing. This process defines an open kinematic chain which approximates the bat skeleton. To identify joint states, the authors

assume that a point fixed rigidly to the l^{th} rigid link can be transformed into the inertial frame by a series of homogeneous transformations which are functions of the joint states and DH model parameters. This approach is the standard method for creating an open kinematic chain skeletal model of a biological system.

One drawback of this method, is that the DH convention requires that *a priori* kinematic model assumptions are made prior to state estimation. Uncertainty in the model makes skeletal joint state estimates inaccurate, and any error incurred while estimating proximal link states is automatically included in the state estimates of all distal links. The goal of this paper is to establish an algorithmic framework for automating the design of robotic joint geometry without requiring these *a priori* assumptions. Thus, we use the methods in [12] to estimate three dimensional locations of individual marker points so that the rigid link assumption does not taint the estimates. The resulting point trajectories are shown in Figure 4.1. Furthermore, we abandon the kinematic models of [13, 14] and assign body

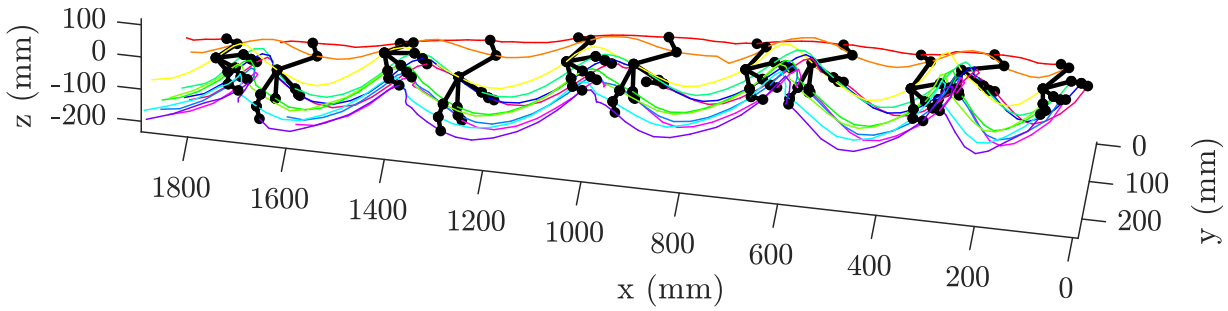


Figure 4.1: Estimated Locations of Feature Points. The colored lines transition from warm colors for points close to the body to cool colors at the figure tips. The black lines depict the skeletal pose every 8 time steps.

fixed frames by using relative orientations of points. This relaxes the constraints between coordinate frames which are imposed by the Denavit-Hartenberg convention. A detailed discussion of how coordinate frames are assigned is presented in Section 4.4.1

4.4 Theoretical Methods

In this section, we present the methodology for using biomotion data for learning joint geometry. We begin by presenting our definition of bat skeletal kinematics. Next, we briefly describe the recent developments in learning theory which enable us to formulate empirical potential functions for the purposes of robotic design. Finally, we discuss how these empirical potential functions are used to design robotic geometry.

4.4.1 Isolating Relative Joint Motion

In this paper, we assign body-fixed coordinate frames using the relative orientations of the proximal and distal points on neighboring pairs of bones. The coordinate frame which is fixed to a particular bone is used to isolate the motion of the point located on the distal end of the bone's distal neighbor. We assume that this motion is caused solely by the joint which connects the two bones. Finally, we assume that all joints have 2 rotational degrees-of-freedom (DOF) and that link length is not fixed when body fixed coordinate frames are assigned. We begin by assigning the humerus-frame and work outwards through the kinematic chain.

To assign the humerus-fixed frame, we assume that the shoulder rotation is responsible for orienting the arm plane which is defined by the shoulder, elbow, and wrist points. The humerus fixed frame is then defined with the $\hat{\boldsymbol{z}}$ direction perpendicular to this plane. The humerus fixed $\hat{\boldsymbol{x}}$ points in the direction of the humerus vector which spans from the shoulder point to the elbow point. Finally, the $\hat{\boldsymbol{y}}$ direction is the cross product of $\hat{\boldsymbol{z}}$ and $\hat{\boldsymbol{x}}$. In other words

$$\boldsymbol{r}_{hum} = \boldsymbol{x}_{elbow} - \boldsymbol{x}_{shoulder}, \tag{4.1}$$

$$\boldsymbol{r}_{rad} = \boldsymbol{x}_{wrist} - \boldsymbol{x}_{elbow}, \tag{4.2}$$

$$\hat{\boldsymbol{z}}_{hum} = \frac{\boldsymbol{r}_{hum} \times \boldsymbol{r}_{rad}}{||\boldsymbol{r}_{hum} \times \boldsymbol{r}_{rad}||}, \tag{4.3}$$

$$\hat{\boldsymbol{x}}_{hum} = \frac{\boldsymbol{r}_{hum}}{||\boldsymbol{r}_{hum}||}, \tag{4.4}$$

$$\hat{\boldsymbol{y}}_{hum} = \hat{\boldsymbol{z}} \times \hat{\boldsymbol{x}}, \tag{4.5}$$

where $\boldsymbol{r}_{hum}$ is the vector which points from the shoulder to the elbow, and $\boldsymbol{r}_{rad}$ is the vector which points from the elbow to the wrist. The three basis vectors, $\hat{\boldsymbol{x}}_{hum}$, $\hat{\boldsymbol{y}}_{hum}$, and $\hat{\boldsymbol{z}}_{hum}$, must have unit magnitudes which is the why the normalizations appear in Equations 4.3 and 4.4. Theoretically, normalization of Equation 4.5 is not required because $\hat{\boldsymbol{x}}_{hum}$ and $\hat{\boldsymbol{z}}_{hum}$ are unit magnitude, however, to avoid numerical precision issues this equation is often normalized in practice. The complete homogeneous transform which describes the humerus fixed frame position and orientation is

$$\mathbf{H}^0_{hum} = \begin{bmatrix} \mathbf{R}^0_{hum} & \boldsymbol{x}_{elbow} \\ \mathbf{0}^T & 1 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}} & \hat{\boldsymbol{y}} & \hat{\boldsymbol{z}} \end{bmatrix}_{hum} & \boldsymbol{x}_{elbow} \\ \mathbf{0}^T & 1 \end{bmatrix} \tag{4.6}$$

where $\mathbf{R}^0_{hum}$ is a rotation matrix describing the relative orientation between the inertial frame and the humerus fixed frame and $\boldsymbol{x}_{elbow}$ is the elbow point. Note that this matrix and all components need to be computed for each time step. Once these matrices are assembled, the motion of the wrist point in the humerus fixed frame is computed as

$$\boldsymbol{x}^{hum}_{wrist} = \mathbf{H}^{hum}_0 \begin{bmatrix} \boldsymbol{x}^0_{wrist} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}^{hum}_0 & -\mathbf{R}^{hum}_0 \boldsymbol{x}_{elbow} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^0_{wrist} \\ 1 \end{bmatrix}. \tag{4.7}$$

where $\mathbf{R}_0^{hum}$ is the matrix-transpose of $\mathbf{R}_{hum}^0$. The transformation in Equation 4.7 is applied in the same manner to each of the remaining distal links after the associated body fixed frames are defined.

To specify the wrist fixed frame, we assume again that the $\hat{\boldsymbol{x}}_{rad}$ direction points along the radius. Then we define the $\hat{\boldsymbol{y}}_{rad}$ direction as the cross product of the $\hat{\boldsymbol{z}}_{hum}$ direction and $\hat{\boldsymbol{x}}_{rad}$. Finally, $\hat{\boldsymbol{z}}_{rad}$ is defined as the cross product of the $\hat{\boldsymbol{x}}_{rad}$ and $\hat{\boldsymbol{y}}_{rad}$ directions. In other words

$$\hat{\boldsymbol{x}}_{rad} = \frac{\boldsymbol{r}_{rad}}{||\boldsymbol{r}_{rad}||}, \tag{4.8}$$

$$\hat{\boldsymbol{y}}_{rad} = \hat{\boldsymbol{z}}_{hum} \times \hat{\boldsymbol{x}}_{rad}, \tag{4.9}$$

$$\hat{\boldsymbol{z}}_{rad} = \hat{\boldsymbol{x}}_{rad} \times \hat{\boldsymbol{y}}_{rad}, . \tag{4.10}$$

These basis vectors are then assembled into rotation matrix $\mathbf{R}_{rad}^0$ and the metacarpal motion is expressed in the radius fixed frame by making the appropriate substitutions into Equation 4.7. Additionally, the remaining body fixed coordinate frames are assigned using the convention described for the metacarpals: the $\hat{\boldsymbol{x}}_{link}$ direction points along the bone, the $\hat{\boldsymbol{y}}_{link}$ direction is perpendicular to the proximal $\hat{\boldsymbol{z}}$, and the new $\hat{\boldsymbol{z}}_{link}$ is the cross product of $\hat{\boldsymbol{x}}_{link}$ and $\hat{\boldsymbol{y}}_{link}$ for the current link. As before, after the associated homogeneous transforms are identified, Equation 4.7 is used to determine the relative motion between the body fixed frame and the end point of its distal neighbor.

Using the transformations described above we can isolate the relative motion between the end point of any bone and the end point of its proximal neighbor. In the future, this data will be used directly to identify geometry of generic joints which permit motion in $SE(3)$. In other words, the joint would allow translation and rotation between the neighboring bodies which, in many biological systems, is more indicative the of the actual motion. For this paper, we simplify the problem and assume that each joint can move in $SO(3)$, and we allow two rotational degrees of freedom, θ_z and θ_y, about the body fixed proximal frame. To identify the rotation about the z-axis

$$\theta_{z,i} = \tan^{-1}\left(\frac{y_i}{x_i}\right) \tag{4.11}$$

where x_i and y_i are the x and y coordinates of the i^{th} sample point. Additionally, the rotation about the y-axis is determined by

$$\theta_{y,i} = \tan^{-1}\left(\frac{z_i}{\sqrt{x_i^2 + y_i^2}}\right) \tag{4.12}$$

where where z_i is the z coordinates of the i^{th} sample point transformed into the proximal body fixed frame. The standard approach for designing robotic hardware typically determines joint

angles in a similar manner. Most-often, box constraints are established on the joint angles. If the identified box constraints are sufficiently narrow for θ_z or θ_y a designer may choose to eliminate that degree of freedom altogether. This approach may overly restrict joint motion. Additionally, if both angles are used in the parameterization, box constraints could enable the joint to reach configurations which were not observed during experiment—such configurations would be observed in the corners of the reachable set. Restricting the motion to the appropriate subspace of the box constraints must then be done in the path planning and control phase of the system design.

In contrast, our approach learns a potential energy field over the joint angles computed from Equations 4.11 and 4.12 and uses this energy field to shape the joint such that only the set of configurations observed during the experiments is possible. The next section presents the foundational theory used to learn a potential energy function from the empirical observations. After learning this potential function, joint geometry is designed to realize the motion.

4.4.2 Identifying the Empirical Potential Functions

Bioinspiring robotic systems from biomotion data is an attempt to design a synthetic system who's dynamics approximate, as closely as possible, the biological system under investigation. One method for describing the dynamics of a system is to derive equations of motion using Lagrangian mechanics. Lagrange's equations for a constrained mechanical system are

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_k}\right) - \frac{\partial T}{\partial q_k} + \frac{\partial U}{\partial q_k} + \frac{\partial \boldsymbol{\phi}}{\partial q_k}\boldsymbol{\lambda}^T = Q_k \tag{4.13}$$

where T is the kinetic energy, U is the potential energy, q_k is the k^{th} state variable, $\boldsymbol{\phi}$ is a vector of holonomic constraints, $\boldsymbol{\lambda}$ is a vector of Lagrange multipliers, and Q_k are the inputs to the system. These equations of motion have been long studied and there exist close form solutions for many mechanical systems. Unfortunately, biological systems are extremely complex, and it is often difficult or impossible to identify these equations of motion for such systems. This is in part why bioinspired robotic systems often approximate the observed biological motion as a collection of ideal rotations subject to box constraints; for such a system, solutions to Lagrange's equations, while non-trivial, can be derived in closed form. However, approximations at this stage of the design process fundamentally prevent design efforts from mimicking biological motion.

Fortunately, recent developments in learning theory have generated methods which identify a motion submanifold from experimental data and use this submanifold to approximate the equations of motion. The Lagrange formulation used in this paper replaces the holonomic constraint term with an empirical potential energy. Thus, Lagrange's equations then become

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_k}\right) - \frac{\partial T}{\partial q_k} + \frac{\partial U}{\partial q_k} + \frac{\partial V_\epsilon}{\partial q_k} = Q_k, \tag{4.14}$$

where V_ϵ is an empirical potential function identified using the experimental observations obtained during biomotion studies. Empirical potential functions are an attractive alternative to holonomic constraints because they can be chosen such that

$$\left[q : V_\epsilon(q;\epsilon) = 0\right] = \left[q : \phi_i(q) = 0, i = 1, \ldots, m\right]. \tag{4.15}$$

In other words, the set of q which make the empirical potential function equal to zero are the same set of q that make the holonomic constraints equal to zero. If we construct the empirical potential function such that the set of q are the experimentally observed joint configurations computed in the previous section, then low potential energy is observed near these experimental observations and high potential energy is observed away from the observations. Thus, deviations from the minimal potential energy set will generate forces that "push" trajectories back toward the set of observed configurations.

An elegant solution for identifying a suitable empirical potential function is provided by the theory of reproducing kernel Hilbert spaces (RKHS). The theory relies on the selection of a kernel function which is real, symmetric, and positive definite. A common choice of kernel function is the radial basis function

$$K(\overline{q}, q) = \exp\left(-\beta||\overline{q} - q||\right) \tag{4.16}$$

where, $\overline{q}$ is the function center, q is the location at which the function is evaluated, and β is a hyper parameter which controls the sharpness of the basis function. In this paper, Equation 4.16 is the kernel with which the empirical potential energy is learned. The derivation of the empirical potential is outside the scope of this paper, so the reader is directed to [28, 31, 88] for a complete discussion. The result, however, is that the empirical potential function is identified as

$$V_\epsilon(q) = \frac{1}{\epsilon^2}\left(1 - \frac{1}{n}K_{\overline{q}}(q)^T g_\lambda(\mathbb{K})K_{\overline{q}}(q)\right). \tag{4.17}$$

where $K_{\overline{q}}(q) := \{K(\overline{q}_1, q), \ldots, K(\overline{q}_n, q)\}^T \in \mathbb{R}^n$, $\mathbb{K}_{st} := K(\overline{q}_s, \overline{q}_t) \in \mathbb{R}^{n\times n}$ for $1 \leq s, t \leq n$, $g_\lambda(\cdot)$ is a filter function as defined in [31], and ϵ is a small parameter which scales the affect of V_ϵ. Appropriate selection of ϵ will be critical in future work for scaling the energy function such that the optimized strain energy of the joint adequately approximates the empirical potential. In this paper, we do not optimize the strain energy of a compliant joint, but rather identify a "zero-potential set" and use its contour to constrain the joint motion. Thus, we choose $\epsilon = 1$ for all energy fields identified in this paper. Finally, $g_\lambda(\cdot)$ as an approximate pseudo-inverse which neglects extreme singular values in $\mathbb{K}$.

Equation 4.17 can be evaluated on any set of experimental data. In this paper, we isolate the motion of a single joint using the method presented in the previous section and identify the empirical potential function for each joint individually. Thus, for a particular joint, the set of sample data is $\mathbf{Q}_{joint} = \{\overline{q}_1, \ldots, \overline{q}_N\}$ where N is the total number of observations, and each $\overline{q}_i = \{\theta_z, \theta_y\}_i^T$ is the i^{th} pair of identified joint angles. In this paper, the assume that each joint is well approximated by two orthogonal rotational degrees of freedom. Thus, ball joints are used as the general joint geometry.

4.4.3 Using the Empirical Potential For Joint Design

The empirical potential functions identified in the last section are used to shape a motion-restricting cap which constrains the motion of the ball joints to the configuration set observed during experiments. Once the empirical potential function is identified, a threshold is applied to the energy field to identify a "zero-energy contour" and an offset is added to allow for the appropriately sized link to protrude from the joint. Because we choose $\epsilon = 1$ in this paper, the energy function will take values between zero and one, and we choose the threshold for the zero potential set to be 0.05. The link radii are 3mm, 0.5mm, and 2mm for the radius, metacarpals, and phalanges, respectively. These radii are chosen due to the tolerances and accuracy which can be achieved using a 3D printer which is used to fabricate all the parts in the bat wing.

The elbow and phalangeal joints consist of five main parts: the distal link, the proximal link, the center ball, the collar, and the end cap. The ball is located at the center of the joint with the collar clipped around the outside. The collar fits into a groove on the proximal link and prevents rotation around the axis pointing down the distal link. The end cap fits over the top of the ball and the contour identified from the empirical potential function is cut out of the end to permit motion on the identified submanifold. Finally, the distal link fits into a hole in the ball and protrudes out of the end cap into the next joint. The assembly for these links is shown in Figure 4.2.

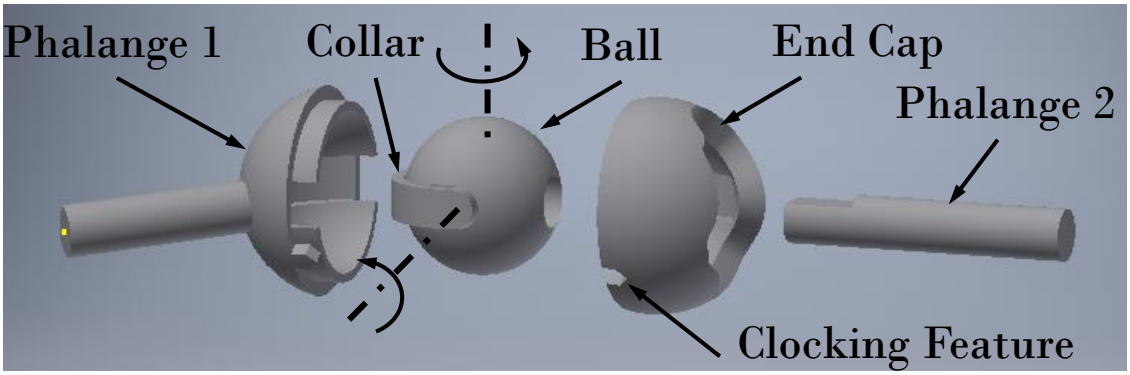


Figure 4.2: Digit 3 Phalange 2 Joint Assembly. Phalange 1 holds the central ball and collar which only allow rotation about the indicated axes (dashed lines). The end-cap is designed using the methods shown above. The clocking feature insures that proper orientation is maintained during assembly.

While the joint in Figure 4.2 is used for most of the wing joints it cannot be used to form the wrist joint because all the metacarpals are assumed to have the same center of rotation. Thus, the center metacarpal has a ball at the proximal end while the outer metacarpals have spherical sections which fit around the ball at their proximal ends. This gives all the joints

the same center of rotation. Instead of using a collar to eliminate the third degree of freedom, pins protrude from the proximal ends of the metacarpals which fit into a track on the back of the radius. Finally, the motion sets of the fourth and fifth metacarpals overlap—which will be shown in the following section. This would cause the identified contours for these links to intersect allowing these digits to occupy the reachable set of the other. To remedy this, a slight rotational offset is applied to the motion data. This separates the motion sets such that independent contours can be generated. After the metacarpal links protrude from the end cap, the offset is reversed to ensure that the final motions of the metacarpals match the original observations.

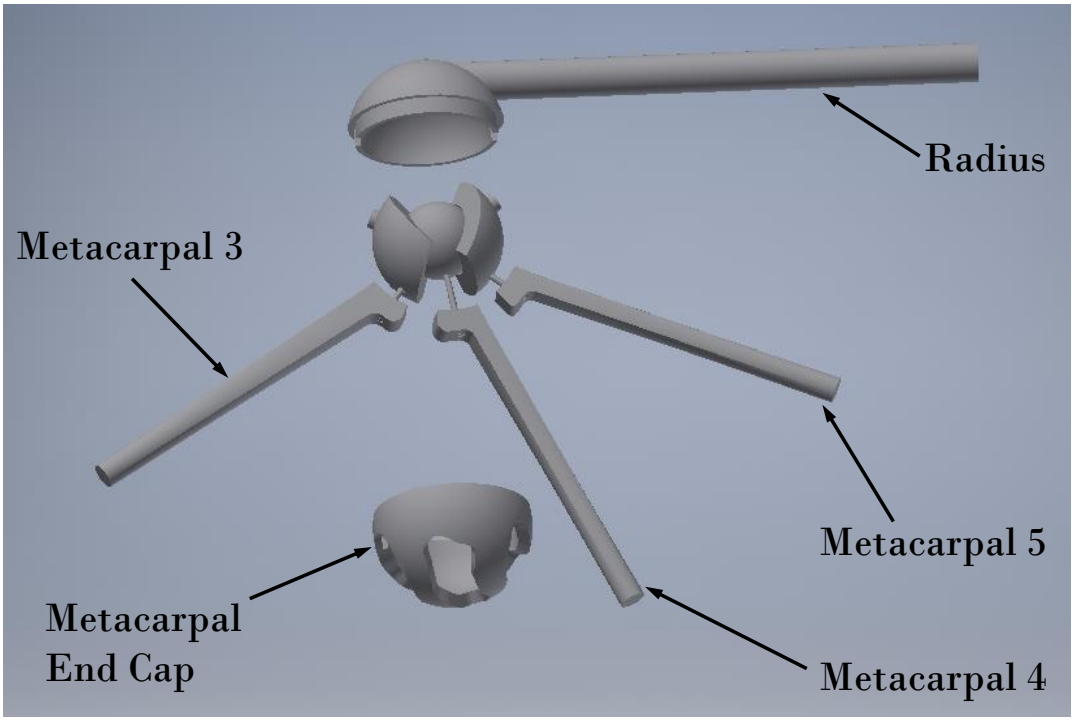


Figure 4.3: Wrist Assembly. The distal end of the radius is shaped like a cup to hold the three metacarpals. The distal ends of metacarpals 3 and 5 are spherical sections which fit over the ball which is located at the end of metacarpal 4. The end cap has range of motion restricting contours for all three metacarpals.

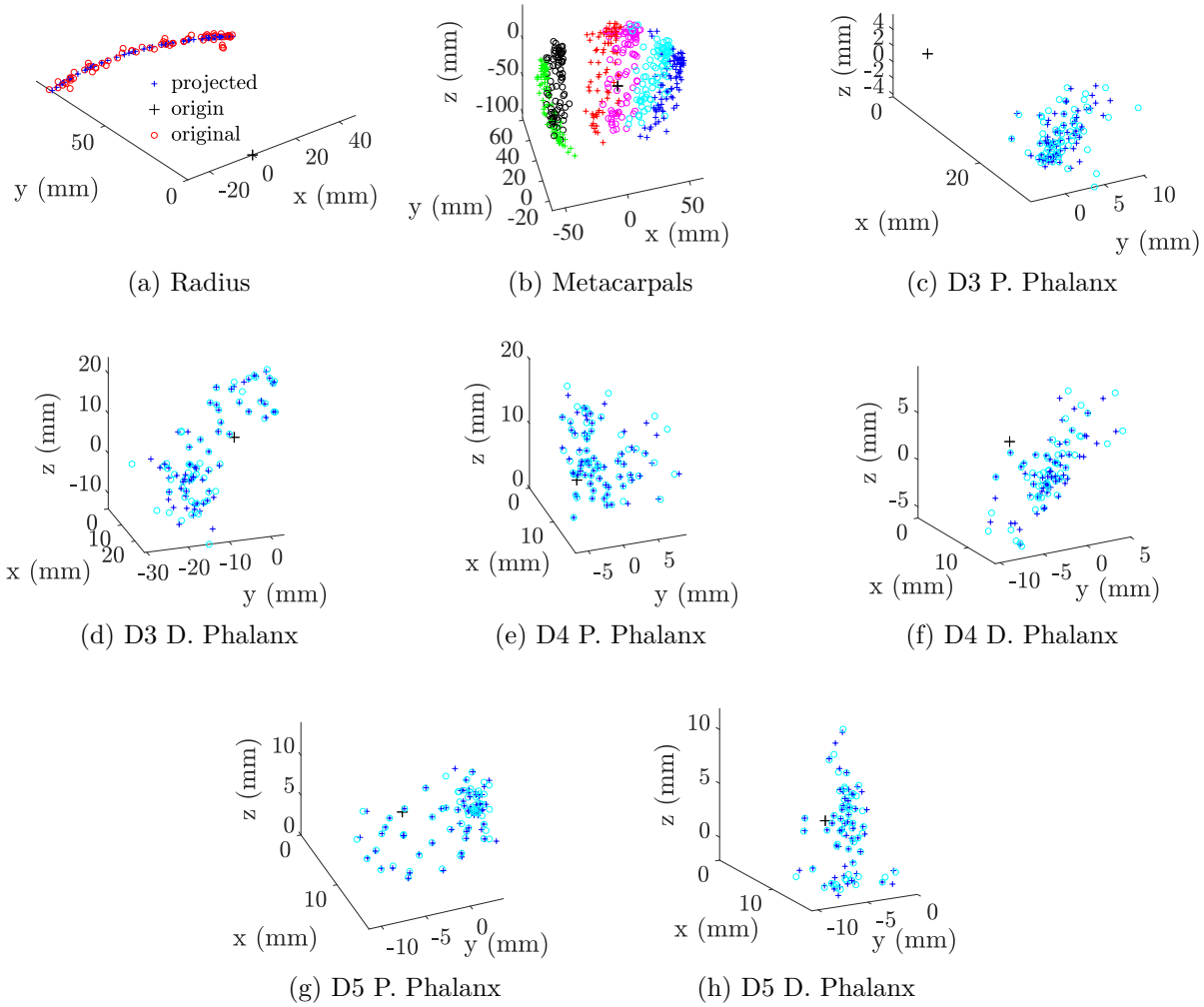
4.5 Results

Using the data from Section 4.3 and the theoretical approach presented in Section 4.4, we isolate motion of each joint, learn the empirical potential functions, identify the zero-potential

set, and generate the contours used to make the final joint geometry which will restrict motion of the fabricated bat wing to the set of configurations observed during experiments.

4.5.1 Kinematic Data

Using the kinematic identification process presented in Section 4.3 we recursively isolate the motion of each link, starting with the radius, and work outward to the distal phalanges. The motion of each individual joint is shown in Figure 4.4.



(a) Radius (b) Metacarpals (c) D3 P. Phalanx

(d) D3 D. Phalanx (e) D4 P. Phalanx (f) D4 D. Phalanx

(g) D5 P. Phalanx (h) D5 D. Phalanx

Figure 4.4: Motion of Wing Bones with Respect to Their Proximal Neighbors. The original data (blue '+') is shown with the data projected onto the surface of a sphere (red 'o'). The relative motion of each bone is dramatically different. Each joint exhibits substantially different motions which will result in dramatically different joint contours.

Due to our identification method, the wrist motion shown in Figure 4.4a is planar and, in this paper, is approximated by a single rotational degree of freedom. However, there is length variation present in the original data which is denoted by the red circles. In a future work, the empirical potential function, and subsequently the fabricated geometry, will be identified to account for this stretching, but in this preliminary work, we only consider rotation.

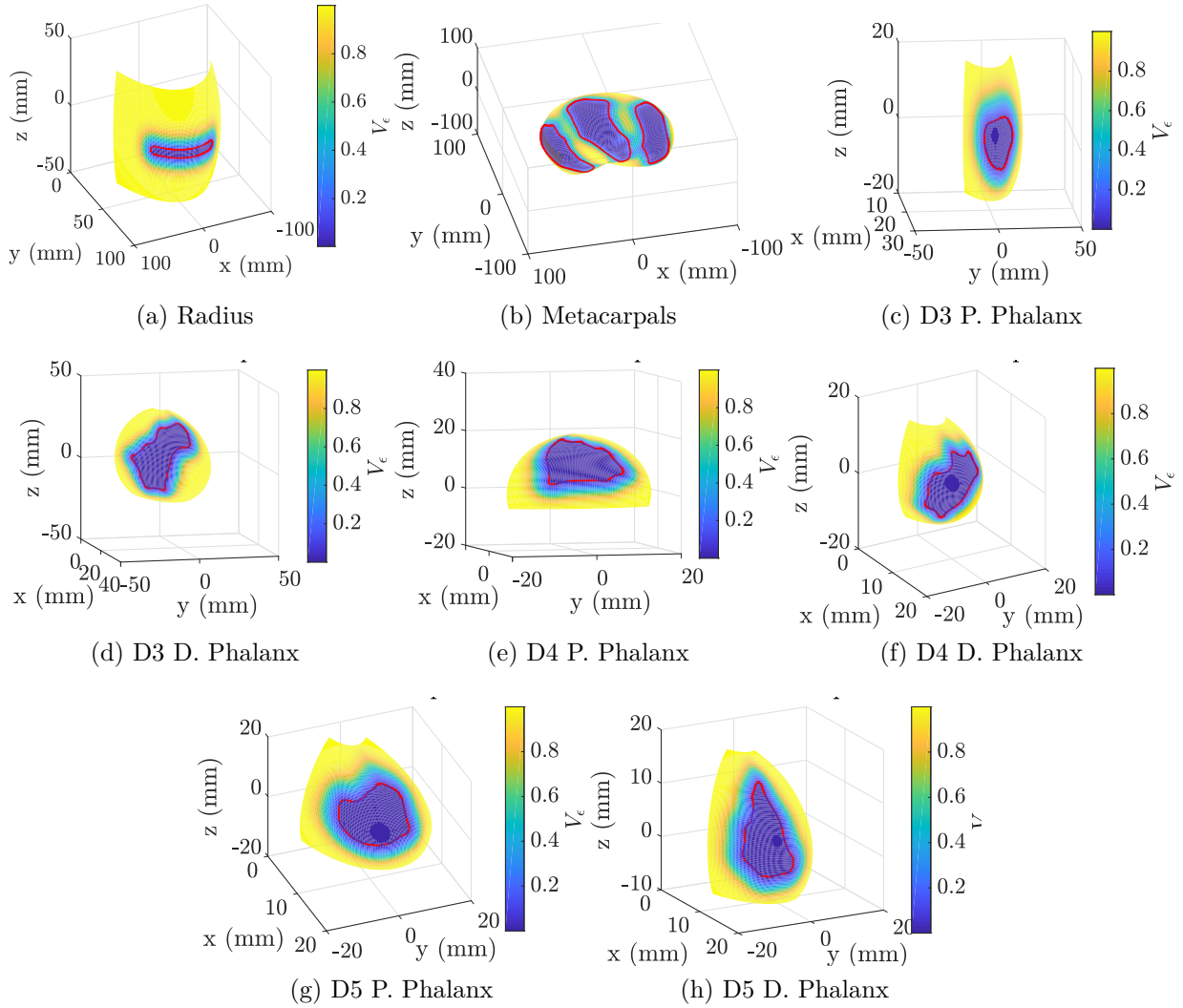
While the motion of the wrist is planar and approximated by a single pin joint, the motion of the metacarpals, shown in Figure 4.4b, is substantially different. The motion of each metacarpal sweeps out an wedge-shaped area on the sphere. This area requires approximately 90 degrees of motion in the first principle direction. When this first direction is in full extension (i.e. $z \approx 0$), more motion is required in the second principle direction than when the first direction is in full flexion. Note also, that the original motion of the fourth and fifth metacarpals (magenta and cyan circles) intersect. If we use this data to design joint geometry, the zero-potential sets will intersect and so will the final contours resulting in a wrist cap which allows these metacarpals to move too far into the reachable set of their neighbor. To remedy this issue, the projected data (green, red, and blue '+' marks) for the third, fourth and fifth phalanges is pre-rotated by 18^o, 14^o, and -15^o, respectively. This prevents the empirical potentials from overlapping which results in distinct contours for each metacarpal opening in the wrist cap. To insure that the motion of the proximal end of each metacarpal is still faithful to the experimental observations, we undo this rotation immediately after the link protrudes from the metacarpal cap by adding a compensating offset in the opposite direction. This produces links which are not correct anatomically, but we are only concerned with motion mimicry in this paper.

While the metacarpal motions are all similar in shape, the motions of the phalanges, which are shown in Figures 4.4c to 4.4h, are distinct from one another. The proximal phalanges tend to have a range of motion which would require approximately the same amount of motion in two principle directions, while the distal phalanges exhibit more motion in the first principle direction than in the second principle direction.

4.5.2 Identified Empirical Potential Functions

Using the kinematic data presented in Figure 4.4 the energy functions for the joints are learned. The parameters used to learn the energy functions are the same for all joints $\lambda = 0.001$ and $\beta = 10$. The learned energy functions are plotted over the sphere and shown in Figure 4.5. In the figure, light colors correspond to high potential energy, and dark colors correspond to low potential energy. Each of these energy functions is used to find the zero-potential (ZP) set assuming a threshold of 0.1. The identified ZP sets are outlined in red.

In the figure, the learned energy functions for all the joints are unique in shape and size. The energy function for the radius is long and thin which is expected due to our specification of the kinematics: the radius is assumed to only move within the plane of the arm. The



(a) Radius (b) Metacarpals (c) D3 P. Phalanx

(d) D3 D. Phalanx (e) D4 P. Phalanx (f) D4 D. Phalanx

(g) D5 P. Phalanx (h) D5 D. Phalanx

Figure 4.5: Learned Energy Functions for Each Joint. Dark colors represent configurations which have low potential energy and lighter colors represent configurations with high potential energy. The red contour is the boundary of the zero potential set.

component of motion which generates samples not on the arc subtended by a fix-length radius link is due to extension in the elbow joint and is not included the empirical potential and, therefore, cannot be included in these visualizations. The energy functions for the two phalanges on the third digit show that little motion is required in either principle direction for the first phalanx, and an arc-like configuration set required for the second phalanx. In contrast, the ZP set for the first phalanx of digit four is triangular in shape while the second phalanx ZP set is similar in shape to that of the third digit second phalanx. Finally, the both phalanges on the fifth digit have notably different motions that the corresponding phalanges on digits three and four. A larger ZP set is required for the first phalanx, and a tear drop

shape ZP set is observed for the second phalanx.

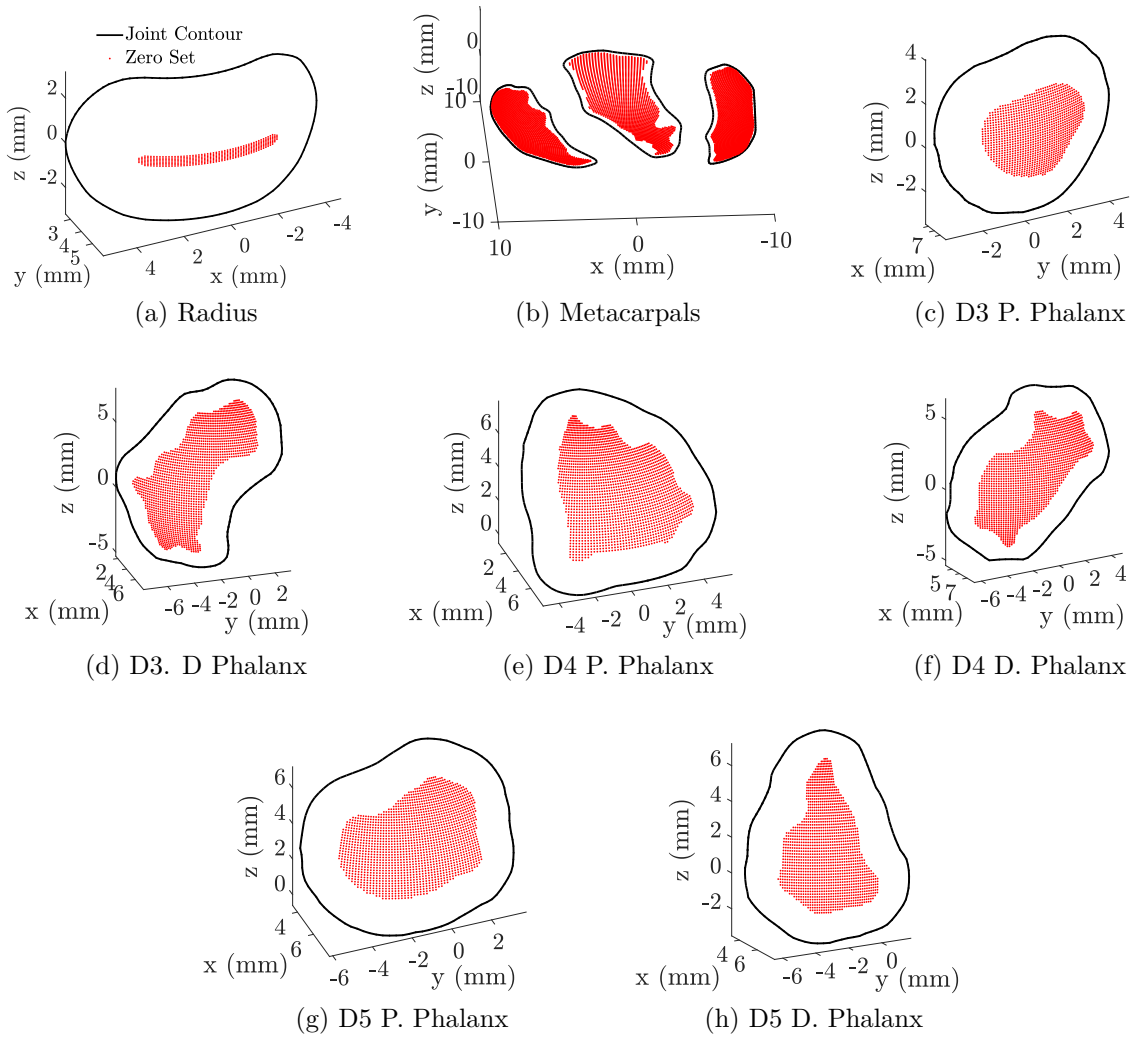
These energy functions show that geometry for each joint must be substantially different to recreate the wing configurations observed during experiment. If box constraints were applied to any of these motions, the links could extend well into the yellow region of the plots which indicates high energy and therefore undesirable, and potentially unnatural configurations. Thus, the ZP sets identified in this section are better suited for generating realistic joint geometry.

4.5.3 Joint Geometry

The ZP sets identified in the last section are used to determine the appropriate contours for constructing the end caps for the joints. As discussed in the last section, the ZP sets indicate the permissible motion of the distal end of each link. The geometry which restricts the motion of each link does so at the proximal end of the link. Thus, the ZP set is projected to a sphere of the same radius as the ball within the joint. Spherical radii are 6mm, 6mm, 8mm, and 6mm for the elbow, fourth metacarpal, third and fifth metacarpal, and phalanges, respectively. Additionally, the ZP set is identified assuming a zero radius link, so the boundary of the ZP set is offset by the radius of the link. Radius offsets are 3mm for the elbow and 0.5mm for the metacarpals, and 2mm for the phalanges. The original ZP sets and the final contours are shown in Figure 4.6.

As expected given the motion data and the empirical potential functions previously presented, the identified contours shown in Figure 4.6 are dramatically different for each joint. The scales of the x, y, and z axes reflect the projection of the ZP sets onto the surface of the ball joint.

As mentioned in the discussion of the empirical potential function, all of the joint contours are identified using fixed λ and β. The choice of these parameters however, can substantially affect the resulting contour. To demonstrate the effects of β on resulting geometry, Figure 4.7 shows the contours generated from the empirical potential function of the second phalanx on digit 4 with β equal to 10, 5, and 1. The figure shows that decreasing β results in substantially smoother contours for the same observed joint motion. This result is expected because the empirical potential function is constructed with radial basis functions centered at each of the observed configurations. The basis functions will become wider with smaller values of β. Smoother contours may be desirable if the experimental data set has a limited number of observations. As with any smoothing or filtering operation, reducing β to may remove artifacts of motion which were important to capture. Figure 4.7 shows that with $\beta = 10$ a non-convex contour is identified, but when $\beta = 1$ the resulting contour becomes convex.



(a) Radius (b) Metacarpals (c) D3 P. Phalanx

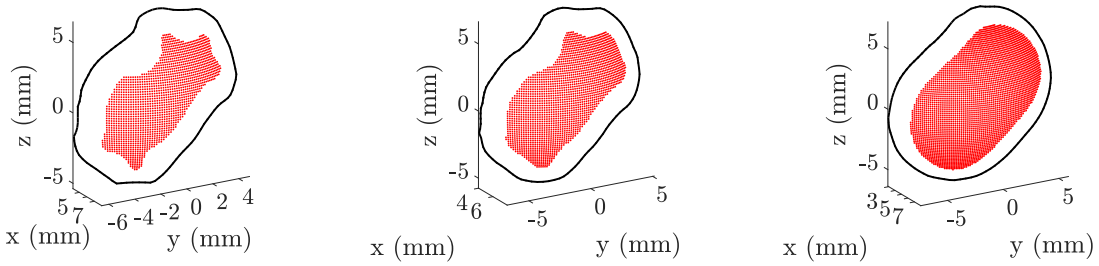
(d) D3. D Phalanx (e) D4 P. Phalanx (f) D4 D. Phalanx

(g) D5 P. Phalanx (h) D5 D. Phalanx

Figure 4.6: Identified Contours for Each Joint. The zero potential set (red dots) and the identified boundary contour (black line) which is used to create the joint geometry.

4.5.4 Physical Prototypes

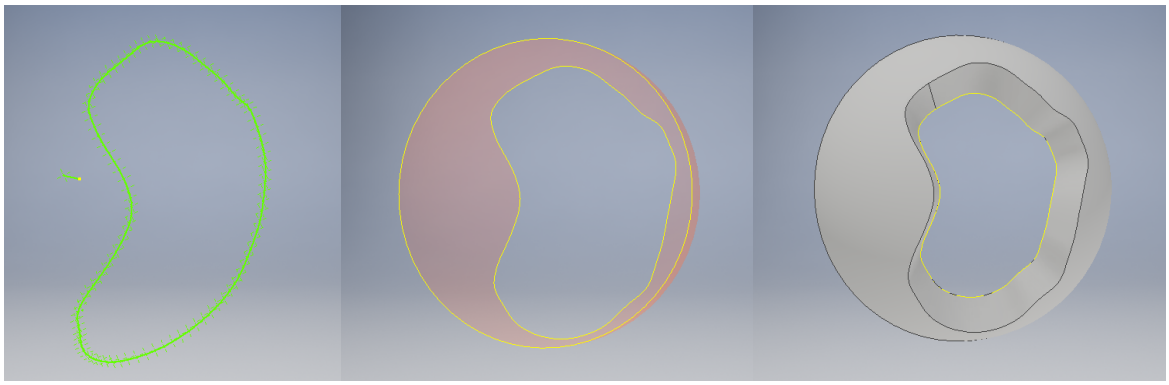
The black contours in Figure 4.6 are the final geometries which are used to generate solid models of the joint end caps. The contours are exported to an Excel™ spread sheet by sampling 100 evenly spaced points from each contour. The points are imported into Autodesk™ Inventor™ and a closed, three-dimensional, interpolation spline is drawn through the points. The spline is then projected onto a hemisphere which is the desired radius of the ball joint plus a tolerance of 0.25mm to insure that the ball can move freely within the resulting joint. Next, the portion of the hemisphere which is inside the contour is deleted. Using the resulting surface, the *Thicken* operation is used to loft material outward and generate the basic geometry. Finally, features are added such that the end cap fits snuggly over the distal end



(a) $\beta = 10$ (b) $\beta = 5$ (c) $\beta = 1$

Figure 4.7: Effect of Hyper-parameter on Contour Shape. As β decreases the contours become smoother.

of the adjoining link and that proper orientation of the two parts is maintained. The basic joint structure of the elbow and all phalangeal joints are the same: a central ball with a collar which restricts motion of the third axis is placed at the center of the spherical joint. The progression from contour import to final fabrication is show in Figure 4.8.

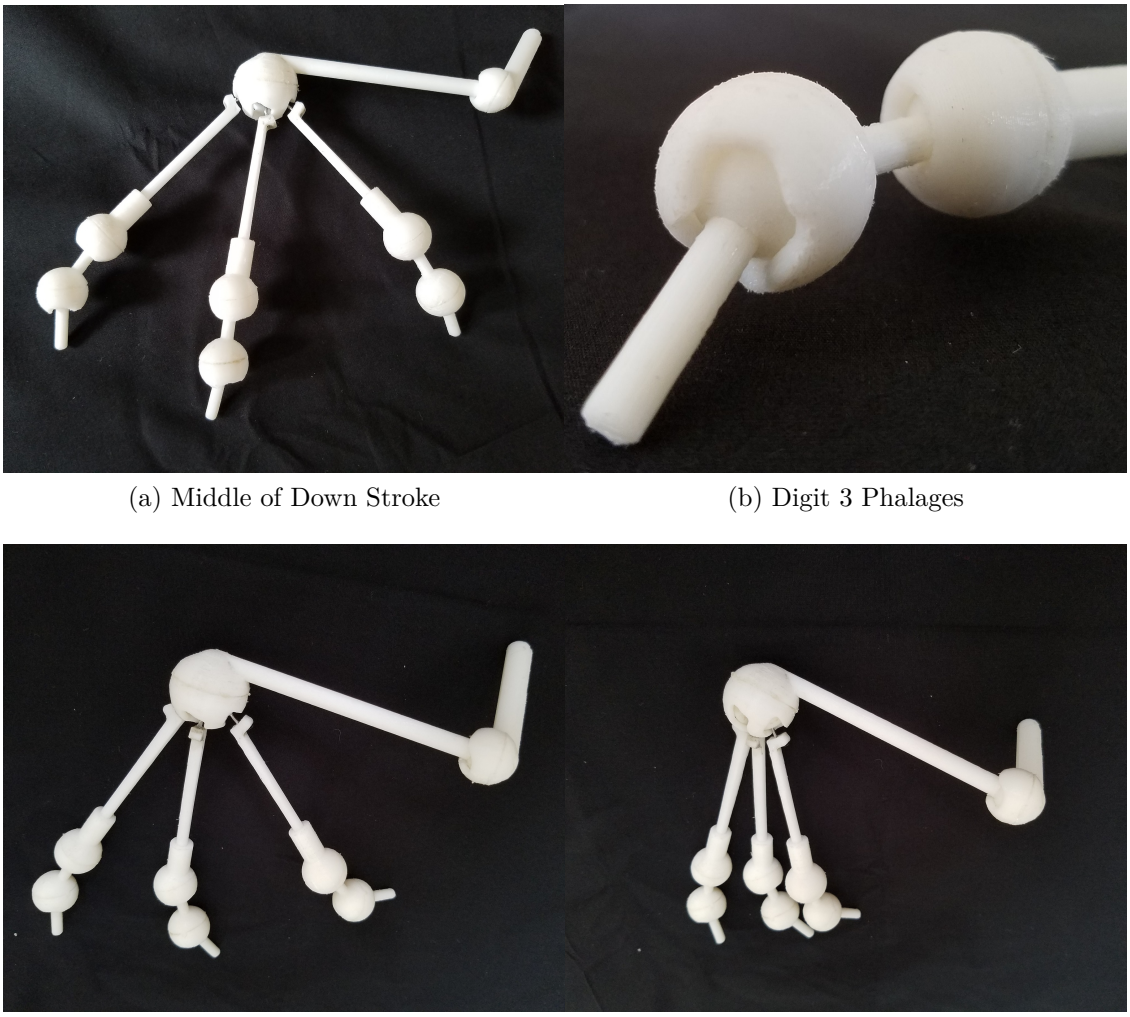


(a) Contour (b) Surface (c) Thickened Geometry

Figure 4.8: Design Process for Elbow and Phalangeal Joints

After modeling is finished, an STL file of each part is exported from Inventor™ and the part is printed on a Monoprice™ Maker Select™ Plus V2 printer. The fully assembled wing is shown in Figure 4.9.

Figure 4.9 shows three skeletal poses of the bat wing and a close-up of the digit 3 phalanges. This realization of the hardware is bulky because of manufacturing constraints imposed by the 3D printer. However, if the joints were made of metal, they could be scaled such that a reasonably sized bat wing could be fabricated and tested in a wing tunnel.



(a) Middle of Down Stroke (b) Digit 3 Phalages

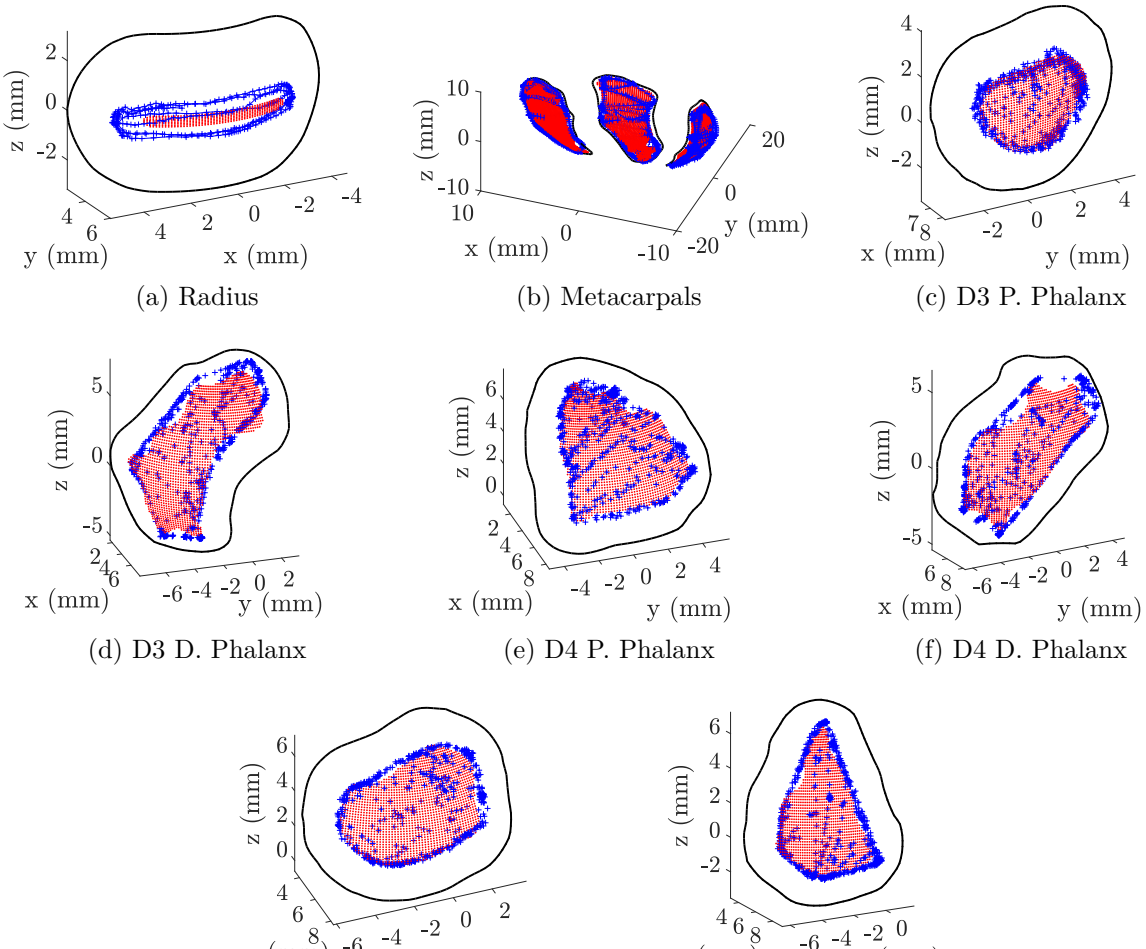
(c) Top of Upstroke (d) Upstroke Pose

Figure 4.9: Bat Wing Designed with Empirical Potential Functions

4.5.5 Submanifold Validation

To insure the zero-potential sets identified in Figure 4.5 are enforced by the contours generated in Figure 4.6 which are realized by the hardware depicted in Figure 4.9, we perform validation testing on each of the joints. The base link of each joint is clamped in a vice and the articulated link is moved through the admissible range of motion. A GoPro Dual Hero system is used as a small baseline stereo pair and an Aperture mini-20d LED photography light is used to illuminate the link during the test. The stereo pair is calibrated using the MatLab stereoCameraCalibrator. A small piece of black felt is glued to the end of the articulated link such that the center can be easily tracked in the resulting images using optical

flow. After calibration and point tracking are complete, stereo triangulations of all sample points are generated and compared to the ZP set. The ZP set is identified at the surface of the ball joint while the three-dimensional validation data is collected at the end of the articulated link. Thus, the 3D validation data is projected back to the surface of the ball joint. The validation data is shown in Figure 4.10.



(a) Radius (b) Metacarpals (c) D3 P. Phalanx

(d) D3 D. Phalanx (e) D4 P. Phalanx (f) D4 D. Phalanx

(g) D5 P. Phalanx (h) D5 D. Phalanx

Figure 4.10: Validation Results for the Fabricated Joints.

The figure shows that all of the joints fabricated exhibit motions close to the zero-potential sets in Figure 4.6. The metacarpal joints exhibit the most error due to the fabrication process with which they are created. As discussed previously, size constraints require that these joints are assembled from three pieces. Angular alignment cannot be guaranteed which is the cause of the offsets in the data. Despite this offset, the shape of each boundary is consistent with the zero potential set. The elbow joint exhibits the second largest deviations from the

identified zero potential set in that the range of motion of this joint is substantially larger than the original ZP set. This is again due to fabrication constraints. Small inconsistencies in the joint end cap are magnified at the end of the 85mm long radius which is where the validation measurements are taken. Finally, the phalangeal joints show only small deviations from the identified ZP set. The deviations are due to subtleties in the generated contours which cannot be fabricated with the printer used in this investigation. From the validation data presented in Figure 4.10 we conclude that our methodology is able to generate robotic joint geometry which restricts motions to the reachable set identified from an analytical empirical potential function learned over experimental observations of biological motion.

4.6 Conclusions

In this paper, we present a novel design paradigm which uses experimental observations of a bat in flight and recent developments in learning theory to identify an empirical potential energy function and learn robotic joint geometry. To do this, we isolate the motion of each joint by assigning body fixed coordinate frames without making the standard rigid link assumptions. Then, we use the relative motion between two joints to identify an energy function which effectively penalizes configuration away from those observed during experiment. Finally, we apply a threshold to the energy function to learn the admissible set of configurations and learn the appropriate geometry for a joint end cap which restricts motions to that set. We validate this method by collecting motion capture data of each joint and show that each joint only permits motions close to the experimental observations. This work serves as the basis for future studies on robotic design.

While the hardware fabricated for this paper is too cumbersome for a free-flying robot, future work will focus on miniaturization, the addition of compliance, coupling between degrees of freedom, placement of actuators, and control strategies which may be derived using the empirical potential function methods identified in this paper.

Chapter 5

Conclusions

The goal of this dissertation is to establish an algorithmic process for synthesizing robotic joint geometry from studies of biological motion. Once the biological system of interest is selected, the methods in this paper can be applied to acquire motion capture data using a highly redundant but low-cost system, process the motion capture data into reliable state estimates despite potentially large-baseline nonlinear motions, learn a low-order dynamic model for the system of interest, and learn an joint geometry for a skeleton which closely mimics the motions of the observed system. In this paper, the process is demonstrated by performing these activities on the complex motions of bats in flight. Furthermore, our adds the following contributions to the state of the art:

1. We use a highly redundant, low cost low frame rate data acquisition system for studying the complex motion of bats in flight.

2. We derive, a conditionally independent Bayes' filter which incorporates inboard state correction to dramatically improve estimation accuracy for the highly nonlinear, large-baseline motions exhibited by bats.

3. We formulate a three degree-of-freedom latent space dynamic model of bat flight using Gaussian Process Dynamic Models—this is the first nonlinear dimensionality reduction of bat flight.

4. We present a methodology for designing and optimized compliant structure around a single degree of freedom joint

5. We formulate a methodology for designing restrictive geometry which automatically designs restrictive end caps for spherical joints.

Previous investigations of bat flight, and may other studies of animal motion for that matter, use high resolution high frame rate cameras for data acquisition. This method enables the use of standard tracking methods such as bootstrapped optical flow or feature descriptor matching for feature correspondence. While expensive cameras make image processing easy, these cameras are prohibitively expensive and may prevent new researchers from entering the field. To remedy this issue, our system of GoPro cameras can be purchased and assembled for an order of magnitude less cost and the associated estimation strategy will yield estimates comparable to that of much more expensive systems.

In addition to reducing data acquisition cost, we also present the first nonlinear dimensionality reduction of bat flight. Our model shows that the flap cycle is described with two independent degrees of freedom and that acceleration is described by a third independent degree of freedom. In the future this model will be used as a motion prior for improving predictions within the state estimator discussed previously in an attempt to automate correspondence.

In addition to deriving a dynamic model of bat flight, this thesis presents two methods for learning joint geometry from motion capture data. Previous investigations have relied on designer interpretation of kinematics data to generate robotic hardware, but due to the proliferation of research in this area a new, automated method is needed. Thus, our framework both designs optimal compliant geometry and optimal rigid geometry which restrict joint motions to an identified manifold. The manifold is identified by using recent developments in learning theory which use analytical empirical energy functions as an analog for traditional honolomic constraints for mechanical systems. The advantage in using the empirical potentials is that no designer interpretation is required to identify the constraining function. Rather, the energy function is built over a set of empirical observations and "pushes" system trajectories to those configurations. We demonstrate the validity of this method by constructing an entire bat wing and presenting validation data which shows that configurations of the fabricated joints remain close to the identified submanifold.

We hope that this thesis provides a foundation for studying biology using low-cost imaging equipment and synthesizing robotic systems directly from the acquired data. To this end, the possibilities for extending this work are almost endless. The data collection methodology can be applied to any system of interest—even one that's not biological. One limitation of our work is that the fabricated joints still exhibit motions on $SO(2)$ and $SO(3)$. Many texts on biology paint a much more complicated picture of the actual kinematics of biological systems. Our framework could be extended to allow 6DOF motion between neighboring links. The hardware would then be optimized as a continuous solid with varying stiffness material. The goal would be to print the entire bat wing—or other biological system—on a variable stiffness three-dimensional printer such that no assembly of the skeleton is required and that the motion mimics, as closely as possible, the motion of the biological system.

In addition to generalizing the joint structure that our methodology accommodates, one could extend this work to identify degree of freedom coupling, optimally place actuators and design realizable controllers. The methods presented herein only consider the relative motion between a link and its proximal or distal neighbor. However, from the GPDM learned in Chapter 2 suggests that joints should move in unison. Therefore, the energy functions identified in Chapters 3 and 4 could be formulated to exploit coupling between non-adjacent links. Finally, the energy fields identified could be differentiated to determine a force field which may be useful in control design or actuator placement. The terminus of this research direction is the complete automation of biologically inspired robotic design.

Bibliography

[1] Ralph Abraham and Jerrold Marsden. *Foundations of Mechanics*. Addison-Wesley Publishing Company, Inc., 1987.

[2] H D Aldridge. Kinematics and aerodynamics of the greater horseshoe bat, Rhinolophus ferrumequinum, in horizontal flight at various flight speeds. *The Journal of Experimental Biology*, 126:479–497, 1986.

[3] Mauricio A Alvarez, Lorenzo Rosasco, and Neil D Lawrence. Kernels for Vector-Valued Functions: a Review. Technical report, Massachusetts Institute of Technology, Cambridge, MA, 2011.

[4] Arvind Ananthanarayanan, Mojtaba Azadi, and Sangbae Kim. Towards a bio-inspired leg design for high-speed running. *Bioinspiration & Biomimetics*, 7(4), 2012.

[5] Moritz Bächer, Bernd Bickel, Doug L. James, and Hanspeter Pfister. Fabricating articulated characters from skinned meshes. *ACM Transactions on Graphics*, 31(4):1–9, 2012.

[6] Joseph W. Bahlman, Rosalyn M. Price-Waldman, Hannah W. Lippe, Kenneth S. Breuer, and Sharon M. Swartz. Simplifying a wing: diversity and functional consequences of digital joint reduction in bat wings. *Journal of Anatomy*, 229(1):114–127, 2016.

[7] Sanaz Bazaz Behbahani and Xiaobo Tan. Bio-inspired flexible joints with passive feathering for robotic fish pectoral fins. *Bioinspiration & Biomimetics*, 11(3):1–14.

[8] Sanaz Bazaz Behbahani and Xiaobo Tan. A flexible passive joint for robotic fish pectoral fins: Design, dynamic modeling, and experimental results. In *IEEE International Conference on Intelligent Robots and Systems*, pages 2832–2838, 2014.

[9] Sanaz Bazaz Behbahani, Jianxun Wang, and Xiaobo Tan. A dynamic model for robotic fish with flexible pectoral fins. In *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics: Mechatronics for Human Wellbeing, AIM 2013*, pages 1552–1557, 2013.

[10] Matt J. Bender, Aishwarya George, Nathan Powell, Andrew Kurdila, and Rolf Müller. Empricial potential functions for driving bioinspired joint design. *ASME Journal of Dynamic Systems, Measurements, and Control*, In Press, 2018.

[11] Matt J. Bender, Jia Guo, Nathan Powell, Andrew Kurdila, and Rolf Müller. Learing bioinspired joint geometry from motion caputre data of bat flight. *Bioinspiration & Biomimetics*, In Review, 2018.

[12] Matt J Bender, Hunter G McClelland, Gerardo Bledt, Andrew Kurdila, Tomonari Furukawa, and Rolf Mueller. Trajectory estimation of bat flight using a multi-view camera system. In *Modeling and Simulation Technologies Conference, SciTech 2015*, pages 1–13, January 2015.

[13] Matt J. Bender, Hunter M. McClelland, Andrew Kurdila, and Rolf Mueller. Recursive bayesian estimation of bat flapping flight using kinematic trees. In *AIAA Modeling and Simulation Technologies Conference*, pages 1–12, January 2016.

[14] Matthew J Bender, Li Tian, Xiaozhou Fan, Andrew Kurdila, and Rolf Müller. Spatially recursive estimation and gaussian process dynamic models of bat flapping fight. *Nonlinear Dynamics*, (In Press), 2018.

[15] Attila J. Bergou, Sharon Swartz, Kenneth Breuer, and Gabriel Taubin. 3D reconstruction of bat flight kinematics from sparse multiple views. *Proceedings of the IEEE International Conference on Computer Vision*, pages 1618–1625, 2011.

[16] Attila J Bergou, Sharon M Swartz, Hamid Vejdani, and Daniel K Riskin. Falling with Style : Bats Perform Complex Aerial Rotations by Adjusting Wing Inertia. pages 1–16, 2015.

[17] Miklós Bergou, Max Wardetzky, Stephen Robinson, Basile Audoly, and Eitan Grinspun. Discrete elastic rods. *ACM Transactions on Graphics*, 27(3):1, 2008.

[18] David Billingsley, Geoff Slipher, Jared Grauer, and James Hubbard. Testing of a Passively Morphing Ornithopter Wing. In *AIAA Infotech@Aerospace Conference*, pages 1–8, April 2009.

[19] A.M. Bloch. *Nonholonomic Mechanics and Control*. Springer, 2003.

[20] Folkmar Bornemann. *Homogenization in Time of Singularly Perturbed Mechanical Systems*, volume 1687 of *Lecture Notes In Mathematics*. Springer-Verlag Berline Heidelberg, 1st edition, 1998.

[21] G Bradski. OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[22] Francesco Bullo and Andrew D Lewis. *Geometric Control of Mechanical Systems*. Springer, 2005.

[23] C Canton-Ferrer, J R Casas, and M Pardas. Towards a low cost multi-camera marker based human motion capture system. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 2581–2584, nov 2009.

[24] Jorn A. Cheney, Daniel Ton, Nicolai Konow, Daniel K. Riskin, Kenneth S. Breuer, and Sharon M. Swartz. Hindlimb motion during steady flight of the lesser dog-faced fruit bat, Cynopterus brachyotis. *PLoS ONE*, 9(5):1–8, 2014.

[25] J Colorado, A Barrientos, C Rossi, and K S Breuer. Biomechanics of smart wings in a bat robot: morphing wings using SMA actuators. *Bioinspiration & Biomimetics*, 8(3):1–16, 2013.

[26] Roy Craig and Andrew Kurdila. *Fundamentals of Structural Dynamics*. John Wiley and Sons, Inc., 2nd edition, 2006.

[27] Utku Çulha and Fumiya Iida. Enhancement of finger motion range with compliant anthropomorphic joint design. *Bioinspiration & Biomimetics*, 11(2):1–16, 2016.

[28] Shirin Dadashi, Hunter G Mcclelland, and Andrew Kurdila. Learning Theory and Empirical Potentials for Modeling Discrete Mechanics. In *IEEE American Control Conference*, number 10, pages 4466–4472, 2017.

[29] L. M. Day and B. C. Jayne. Interspecific scaling of the morphology and posture of the limbs during the locomotion of cats (Felidae). *Journal of Experimental Biology*, 210(4):642–654, 2007.

[30] Fernando De La Torre, Jessica Hodgins, Adam W. Bargteil, Xavier Martin, Justin C. Macey, Alex Collado, and Pep Beltran. Guide to the Carnegie Mellon University Multimodal Activity (CMU-MMAC) Database. Technical Report April, Carnegie Mellon University, Pittsburgh, Pennsylvania, 2008.

[31] Ernesto De Vito, Lorenzo Rosasco, and Alessandro Toigo. Learning Sets with Separating Kernels. *Applied and Computational Harmonic Analysis*, 37(2):185–217, 2014.

[32] Frank Dellaert, S.M. Seitz, Sebastian Thrun, and Charles Thorpe. Feature Correspondence : A Markov Chain Monte Carlo Approach. *Advances in Neural Information Processing Systems*, pages 852–858, 2001.

[33] Marcin Eichner and Vittorio Ferrari. Better appearance models for pictorial structures. *Procedings of the British Machine Vision Conference (BMVC)*, pages 1–11, 2009.

[34] A Erol, G Bebis, M Nicolescu, R D Boyle, and X Twombly. A Review on Vision-Based Full DOF Hand Motion Estimation. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (June):75–81, 2005.

[35] Roy Featherstone. *Rigid Body Dynamics Algorithms*. Springer, Berlin, 2008.

[36] Alice C Gibb, Bruce C Jayne, and George V Lauder. Kinematics of Pectoral Fin Locomotion in the Bluegill Sunfish Lepomis Macrochirus. *J. Exp. Biol*, 189:133–161, 1994.

[37] Herbert Goldstein, Charles P Poole, and John Safko. *Classical Mechanics*. Pearson, 3rd edition, 2011.

[38] Pradeep Gopalakrishnan and Danesh K. Tafti. Effect of Wing Flexibility on Lift and Thrust Production in Flapping Flight. *AIAA Journal*, 48(5):865–877, 2010.

[39] Donald T Greenwood. *Principles of Dynamics*. Prentice-Hall, Inc., 2nd edition, 1988.

[40] Ralph Gross and Jianbo Shi. The cmu motion of body (mobo) database. Technical Report CMU-RI-TR-01-18, Carnegie Mellon University, Pittsburgh, PA, June 2001.

[41] Ernst Hairer, Christian Lubich, and Gerhard Wanner. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer, 2nd edition, 2006.

[42] Søren Hauberg. *Spatial Models of Human Motion*. PhD thesis, University of Copenhagen, October 2011.

[43] Søren Hauberg, François Lauze, and Kim Steenstrup Pedersen. Unscented Kalman Filtering on Riemannian Manifolds. *Journal of Mathematical Imaging and Vision*, 46(1):103–120, aug 2013.

[44] Tyson L Hedrick. Software techniques for two- and three-dimensional kinematic measurements of biological and biomimetic systems. *Bioinspiration & Biomimetics*, 3(3):034001, 2008.

[45] J. Hoff, A. Ramezani, S. Chung, and S. Hutchinson. Synergistic design of a bio-inspired micro aerial vehicle with articulated wings. In *Proc. Robotics: Science and Systems*, pages 1–9, 2016.

[46] M. Hofmann and D. M. Gavrila. Multi-view 3D human pose estimation in complex environment. *International Journal of Computer Vision*, 96(1):103–124, 2012.

[47] Michael B. Holte, Cuong Tran, Mohan M. Trivedi, and Thomas B. Moeslund. Human pose estimation and activity recognition from multi-view videos: Comparative explorations of recent developments. *TIEEE Journal of Selected Topics in Signal Processing*, 6(5):538–552, 2012.

[48] T Y Hubel, D K Swartz, and K S Breuer. Wake Structure and Wing Kinematics: The Flight of the Lesser Dog-Faced Fruit Bat, \emph{Cynopterus brachyotis}. *J. Exp. Biol.*, 213:3427–3440, 2010.

[49] Tatjana Y. Hubel, Nickolay I. Hristov, Sharon M. Swartz, and Kenneth S. Breuer. Time-resolved wake structure and kinematics of bat flight. *Experimental Fluids*, pages 933–943, 2010.

[50] Penny E. Hudson, Sandra A. Corr, Rachel C. Payne-Davis, Sinead N. Clancy, Emily Lane, and Alan M. Wilson. Functional anatomy of the cheetah (Acinonyx jubatus) hindlimb. *Journal of Anatomy*, 218(4):363–374, 2011.

[51] Daniel J. Inman. *Engineering Vibrations*. Pearson Education, Inc., Upper Saddle River, New Jersey, 3rd edition, 2008.

[52] J Irarte-Diaz, Daniel K Riskin, D J Willis, K S Breuer, and S M Swartz. Whole-Body Kinematics of a Fruit Bat Reveal the Influence of Wing Inertia on Body Accelerations. *J. Exp. Biol.*, 214:1546–1553, 2011.

[53] José Iriarte-Díaz and S M Swartz. Kinematics of slow turn maneuvering in the fruit bat *Cynopterus Brachyotis*. *J. Exp. Biol*, 211:3478–3489, 2008.

[54] Michael Isard and Andrew Blake. ICondensation: Unifying low-level and high-level tracking in a stochastic framework. In *5th European Conference on Computer Vision*, pages 893–908. IEEE, 1998.

[55] S Kajita, K Yokoi, H Hirukawa, and K Harada. Humanoid Robot HRP-3. In *IEEE International Conference on Intelligent Robots and Systems*, pages 2471–2478, Nice, France, 2008. IEEE.

[56] Dana Kulić, Gentiane Venture, Katsu Yamane, Emel Demircan, Ikuo Mizuuchi, and Katja Mombaur. Anthropomorphic Movement Analysis and Synthesis: A Survey of Methods and Applications. *IEEE Transactions on Robotics*, 32(4):1–20, 2016.

[57] Pei-Hsin Kuo. *Bio-inspired Robotic Joint and Manipulator : from Biomechanical Experimentation and Modeling to Human-like Compliant Finger Design and Control*. Phd, University of Texas at Austin, 2015.

[58] George V. Lauder. Fish Locomotion: Recent Advances and New Directions. *Annual Review of Marine Science*, 7(1):521–545, 2015.

[59] N.D. Lawrence. Gaussian process latent variable models for visualization of high dimensional data. In *NIPS*, number 5, pages 329–336, 2003.

[60] Jie Lei, Mingli Song, Ze Nian Li, and Chun Chen. Whole-body humanoid robot imitation with pose similarity evaluation. *Signal Processing*, 108:136–146, 2015.

[61] Zhao Liu, Jianke Zhu, Jiajun Bu, and Chun Chen. A survey of human pose estimation: The body parts parsing based methods. *Journal of Visual Communication and Image Representation*, 32:10–19, 2015.

[62] F. Lotti, P. Tiezzi, G. Vassura, L. Biagiotti, G. Palli, and C. Melchiorri. Development of UB Hand 3: Early results. *Proceedings - IEEE International Conference on Robotics and Automation*, 2005(April):4488–4493, 2005.

[63] F. Lotti and G. Vassura. A novel approach to mechanical design of articulated fingers for robotic hands. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1687–1692, October 2002.

[64] Yi Ma, Stefano Soatto, Jana Kosecká, and S Shankar Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models*. Springer, 1st edition, 2004.

[65] J MacCormick and M Isard. Partitioned Sampling, Articulated Objects, and Interface-quality Hand Tracking. In *6th European Conference on Computer Vision*, pages 3–19, 2000.

[66] David J C Mackay. *Information Theory , Inference , and Learning Algorithms*. Cambridge University Press, Cambridge, England, 7.2 edition, 2003.

[67] Ceccarelli Marco and Carbone Giuseppe. A new leg design with parallel mechanism architecture. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, pages 1447–1452, 2009.

[68] Jerrold E Marsden and Matthew West. Discrete Mechanics and Variational Integrators. *Acta Numerica*, 10:357–514, may 2001.

[69] Leonard Meirovitch. *Methods of Analytical Dynamics*. Dover, 2003.

[70] Ikuo Mizuuchi, Yuto Nakanishi, Yoshinao Sodeyama, Yuta Namiki, T. Nishino, Naoya Muramatsu, Junichi Urata, Kazuo Hongo, Tomoaki Yoshikai, and Masayuki Inaba. An advanced musculoskeletal humanoid kojiro. *Proceedings of the 2007 7th IEEE-RAS International Conference on Humanoid Robots, HUMANOIDS 2007*, pages 294–299, 2008.

[71] Thomas B. Moeslund and Erik Granum. A Survey of Computer Vision-Based Human Motion Capture. *Computer Vision and Image Understanding*, 81(3):231–268, 2001.

[72] Thomas B. Moeslund, Adrian Hilton, and Volker Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2–3):90–126, 2006.

[73] J Cortes Monforte. *Geometric, Control and Numerical Aspects of Nonholonomic Systems*, volume 10. Springer-Verlag, 2002.

[74] Dominik Mueller and John W Gerdes. Incorporation of passive wing folding in flapping wing miniature air vehicles. *Mechanical Engineering*, 2009.

[75] Gabe Nelson, Aaron Saunders, Neil Neville, Ben Swilling, Joe Bondaryk, Devin Billings, Chris Lee, Robert Playter, and Marc Raibert. PETMAN: A Humanoid Robot for Testing Chemical Protective Clothing. *Journal of the Robotics Society of Japan*, 30(4):372–377, 2012.

[76] R Niiyama and Y. Kuniyoshi. Design of a Musculoskeletal Athlete Robot: A Biomechanical Approach. In *International Conference on Climbing and Walking Robots (CLAWAR)*, pages 173–180, 2009.

[77] Christopher T Orlowski and Anouck R Girard. Modeling and Simulation of Nonlinear Dynamics of Flapping wing Micro Air Vehicles. *AIAA Journal*, 49(5):969–981, 2011.

[78] Ill Woo Park, Jung Yup Kim, Jungho Lee, and Jun Ho Oh. Mechanical design of humanoid robot platform khr-3 (kaist humanoid robot - 3: Hubo). In *Proceedings of 2005 5th IEEE-RAS International Conference on Humanoid Robots*, pages 321–326, 2005.

[79] Jesús Pérez, Bernhard Thomaszewski, Stelian Coros, Bernd Bickel, José A. Canabal, Robert Sumner, and Miguel A. Otaduy. Design and fabrication of flexible rod meshes. *ACM Transactions on Graphics*, 34(4):138:1–138:12, 2015.

[80] Alireza Ramezani, Soon-jo Chung, and Seth Hutchinson. A biomimetic robotic platform to study flight specializations of bats. *Science Robotics*, 2:1–13, 2017.

[81] Alireza Ramezani, Xichen Shi, Soon-jo Chung, and Seth Hutchinson. Lagrangian Modeling and Flight Control of Articulated-Winged Bat Robot. In *IEEE International Conference on Intelligent Robots and Systems*, pages 2867–2874, 2015.

[82] Alireza Ramezani, Xichen Shi, Soon Jo Chung, and Seth Hutchinson. Bat Bot (B2), a biologically inspired flying machine. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 3219–3226, 2016.

[83] Alireza Ramezani, Xichen Shi, Soon-Jo Chung, and Seth Hutchinson. Nonlinear Flight Controller Synthesis of a Bat-Inspired Micro Aerial Vehicle. In *AIAA Guidance, Navigation, and Control Conference*, pages 1–14, January 2016.

[84] Carl E. Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. Adaptative computation and machine learning series. University Press Group Limited, 2006.

[85] Daniel K Riskin, a. Bergou, K. S. Breuer, and S. M. Swartz. Upstroke wing flexion and the inertial cost of bat flight. *Proceedings of the Royal Society B: Biological Sciences*, 279(1740):2945–2950, 2012.

[86] Daniel K Riskin, José Iriarte-Díaz, K M Middleton, K S Breuer, and S M Swartz. The effect of body size on the wing movements of pteropodid bats, with insights into thrust and lift production. *The Journal of Experimental Biology*, 213:4110–4122, 2010.

[87] Daniel K. Riskin, David J. Willis, José Iriarte-Díaz, Tyson L. Hedrick, Mykhaylo Kostandov, Jian Chen, David H. Laidlaw, Kenneth S. Breuer, and Sharon M. Swartz. Quantifying the complexity of bat wing kinematics. *Journal of Theoretical Biology*, 254(3):604–615, 2008.

[88] Lorenzo Rosasco, Mikhail Belkin, and Ernesto De Vito. On Learning with Integral Operators. *Journal of Machine Learning Research*, 11(March):905–934, 2010.

[89] Andre Rosendo, Shogo Nakatsu, Kenichi Narioka, and K O H Hosoda. Producing alternating gait on uncoupled feline hindlimbs: muscular unloading rule on a biomimetic robot. *Advanced Robotics*, 28(6):351–365, 2014.

[90] Wolfgang Send, Markus Fischer, Kristof Jebens, Rainer Mugrauer, Agalya Nagarathinam, and Felix Scharstein. Artificial hinged-wing bird with active torsion and partially linear kinematics. *28th Congress of the International Council of the Aeronautical Sciences*, pages 23–28, 2012.

[91] Sangok Seok, Albert Wang, Meng Yee Chuah, Dong Jin Hyun, Jongwoo Lee, David M. Otten, Jeffrey H. Lang, and Sangbae Kim. Design principles for energy-efficient legged locomotion and implementation on the MIT Cheetah robot. *IEEE/ASME Transactions on Mechatronics*, 20(3):1117–1129, 2015.

[92] W. Shyy, H. Aono, S.K. Chimakurthi, P. Trizila, C.-K. Kang, C.E.S. Cesnik, and H. Liu. Recent progress in flapping wing aerodynamics and aeroelasticity. *Progress in Aerospace Sciences*, 46(7):284–327, oct 2010.

[93] Leonid Sigal, Alexandru O. Balan, and Michael J. Black. HumanEva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International Journal of Computer Vision*, 87(1-2):4–27, 2010.

[94] Leonid Sigal, Michael Isard, Horst Haussecker, and Michael J. Black. Loose-limbed people: Estimating 3D human pose and motion using non-parametric belief propagation. *International Journal of Computer Vision*, 98(1):15–48, 2012.

[95] Mélina Skouras, Bernhard Thomaszewski, Stelian Coros, Bernd Bickel, and Markus Gross. Computational design of actuated deformable characters. *ACM Transactions on Graphics*, 32(4):1–9, 2013.

[96] Mark W. Spong, Seth Hutchinson, and Vidyasagar M. *Robot Modeling and Control*. Number 1. John Wiley and Sons, Hoboken, NJ, 1st edition, 2006.

[97] B Stenger, A Thayananthan, P H S Torr, and R Cipolla. Filtering Using a Tree-Based Estimator. *ICCV '03 Proceedings of the Ninth IEEE International Conference on Computer Vision*, 2:1063–1071, 2003.

[98] Bjorn Stenger, Arasanathan Thayananthan, Philip H S Torr, and Roberto Cipolla. Model-based hand tracking using a hierarchical bayesian filter. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9):1372–1384, 2006.

[99] Mao Sun. Insect flight dynamics: Stability and control. *Reviews of Modern Physics*, 86(2):615–646, 2014.

[100] T Svoboda, D Martinec, and T Pajdla. A convenient multicamera self-calibration for virtual environments. *PRESENCE: Teleoperators and Virtual Environments*, 14(4):407–422, August 2005.

[101] S M Swartz. Mechanical Properties of Bat Wing Membrane Skin. *Zoology*, 239:357–378, 1996.

[102] S.M. Swartz and N. Konow. Advances in the study of bat flight: the wing and the wind. *Canadian Journal of Zoology*, 93:977–990, 2015.

[103] Richard Szeliski. *Computer Vision: Algorithms and Applications*. Springer, London, England, 2004.

[104] A. Thayananthan, B. Stenger, P. Torr, and R. Cipolla. Learning a Kinematic Prior for Tree-Based Filtering. *Procedings of the British Machine Vision Conference 2003*, pages 1–10, 2003.

[105] The Mathworks Inc. Computer Vision System Toolbox, 2014.

[106] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. MIT press, London, England, 2005.

[107] Xiaodong Tian, José Iriarte-Díaz, Kevin Middleton, Ricardo Galvao, Emily Israeli, Abigail Roemer, Allyce Sullivan, Arnold Song, Sharon Swartz, and Kenneth Breuer. Direct measurements of the kinematics and dynamics of bat flight. *Bioinspiration & Biomimetics*, 1(4):S10–S18, 2006.

[108] Raquel Urtasun, David J Fleet, and Pascal Fua. Gaussian Process Dynamical Models for 3D People Tracking. In *IEEE Computer Science Conference on Computer Vision and Patern Recognition*, pages 238–245, 2006.

[109] Raquel Urtasun, David J Fleet, and Neil D Lawrence. Modeling human locomotion with topologically constrained latent variable models. In *Human Motion – Understanding, Modeling, Capture and Animation*, pages 104–118, Rio de Janeiro, Brazil, 2007.

[110] Björn Verrelst, Ronald Van Ham, Bram Vanderborght, Frank Daerden, Dirk Lefeber, and Jimmy Vermeulen. The pneumatic biped "lucy" actuated with pleated pneumatic artificial muscles. *Autonomous Robots*, 18(2):201–213, 2005.

[111] K. Viswanath and K. Nagendra. Climbing Flight of a Fruit Bat Deconstructed. *52nd Aerospace Sciences Meeting, 13-17 January 2014*, (January):1–24, 2014.

[112] K. Viswanath, K. Nagendra, J. Cotter, M. Frauenthal, and D. K. Tafti. Straight-line climbing flight aerodynamics of a fruit bat. *Physics of Fluids*, 26(2), 2014.

[113] Kamal Viswanath and Danesh K. Tafti. Effect of Frontal Gusts on Forward Flapping Flight. *AIAA Journal*, 48(9):2049–2062, 2010.

[114] Jack Wang, David Fleet, and Aaron Hertzmann. Gaussian process dynamical models. *Advances in Neural Information Processing Systems*, pages 1441–1448, 2006.

[115] Jack M Wang, David J Fleet, and Aaron Hertzmann. Gaussian Process Dynamic Models MatLab Code, 2008.

[116] Jack M Wang, David J Fleet, and Aaron Hertzmann. Gaussian Process Dynamical Models for Human Motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):283–298, 2008.

[117] Holger Wendland. *Scattered Data Approximation*. Cambridge University Press, 2013.

[118] Aimy Wissa, Nelson Guerreiro, Jared a. Grauer, James Hubbard, Mary Frecker, Cornelia Altenbuchner, Yashwanth Tummala, and Richard Roberts. Flight Testing of Novel Compliant Spines for Passive Wing Morphing on Ornithopters. *54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, pages 1–17, 2013.

[119] Marta Wolf, L Christoffer Johansson, Rhea von Busse, York Winter, and Anders Hedenström. Kinematics of flight and the relationship to the vortex wake of a Pallas' long tongued bat (Glossophaga soricina). *The Journal of Experimental Biology*, 213(12):2142–2153, 2010.

[120] Robert J. Wood. The first takeoff of a biologically inspired at-scale robotic insect. *IEEE Transactions on Robotics*, 24(2):341–347, 2008.

[121] Xiaojun Zhao, Qiang Huang, Zhaoqin Peng, and Kejie Li. Kinematics mapping and similarity evaluation of humanoid motion based on human motion capture. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, volume 1, pages 840–845, 2004.