VISUALIZING GEOSPATIAL UNCERTAINTY IN MARINE ANIMAL TRACKS

by

Maswood Hasan Mostafi

Submitted in partial fulfillment of the
requirements for the degree of
Master of Computer Science

at

Dalhousie University
Halifax, Nova Scotia
April 2011

DALHOUSIE UNIVERSITY

FACULTY OF COMPUTER SCIENCE

The undersigned hereby certify that they have read and recommended to the Faculty of Graduate Studies for acceptance a thesis entitled "VISUALIZING GEOSPATIAL UNCERTAINTY IN MARINE ANIMAL TRACKS" by Maswood Hasan Mostafi in partial fulfillment of the requirements for the degree of Master of Computer Science.

Dated: April 12, 2011

Supervisors: _____

_____

Reader: _____

DALHOUSIE UNIVERSITY

DATE: April 12, 2011

AUTHOR: Maswood Hasan Mostafi

TITLE: VISUALIZING GEOSPATIAL UNCERTAINTY IN MARINE
ANIMAL TRACKS

DEPARTMENT OR SCHOOL: Faculty of Computer Science

DEGREE: M.C.Sc.   CONVOCATION: October   YEAR: 2011

_____

Signature of Author

Table of Contents

List of Tables

List of Figures

Abstract

Electronically collected animal movement data has been analyzed either statistically or visually using generic geographical information systems. The area of statistical analysis in this field has made progress over the last decade. However, visualizing the movement and behavior remains an open research problem. We have designed and implemented an interactive visualization system, MarineVis, to visualize geospatial uncertainty in the trajectories of marine animals. Using MarineVis, researchers are able to access, analyze and visualize marine animal data and oceanographic data with a variety of approaches. In this thesis, we discuss the MarineVis design structure, rendering techniques, and other visualization techniques which are used by existing software such as IDV to which we compare and contrast the visualization features of our system. Finally, directions of future work related to MarineVis are proposed which will inspire others to further study the challenging but amazingly interesting and exciting research field of marine visualization.

List of Abbreviations and Symbols Used

Acknowledgements

I would like to express my sincere thanks to both of my supervisors, Dr. Dirk Arnold and Dr. Stephen Brooks, for their continual guidance, help and suggestions to solve the problems we have encountered during this research. I also thank Dalhousie University and the Faculty of Computer Science for helping make my research enjoyable and inspiring.

In addition, throughout my research I have received suggestions from different people. Special thanks are given Prof. Ian D. Jonsen and Trevor D. Davies from Bedford Oceanographic Institute for providing the turtle data and NetCDF files as well as giving feedback towards our research from time to time. I would also like to thank Don Murray, Tom Whittaker, and Jeff McWhirter of Unidata Center for helping us fix errors in NetCDF files. My thanks go to Stuart Wier of Unidata as well, for helping us to become familiar with IDV at the initial stage of our research.

I would also like to thank Dr. Norman Scrimger for his valuable suggestions to improve the writing of this thesis.

Finally my parents, Mostafa Roushan and Masrura Roushan, deserve endless thanks for their patience and support.

Chapter 1

Introduction

1.1   Problem Statement

There is a pressing need to integrate oceanographic information with electronic tracking data to understand marine animals' interactions with their environment. Marine animals spend their lives in an environment that for the most part is invisible to humans. Some species spend prolonged periods of time either migrating or foraging and often only ten percent of an animal's time is spent at the surface [7]. Marine scientists, biologists as well as researchers from other disciplines are seeking answers to questions such as:

1. How do marine animals interact with the ocean and how do these interactions affect their migrating and/or foraging behavior?

2. Why do animals migrate over a long distance and/or remain in a particular part of the ocean for a significant duration?

3. Why do marine animals return to certain places more frequently than others? Are there environmental factors behind this, or is it due to their nature?

4. How do other factors such as ocean water temperature, salinity, and velocity affect the lives of different marine animals in the ocean? Are there particular types of species that might become extinct over a short or a long period of time?

5. Depending on the tracks of these marine animals, how should shipping regulations be altered? Is there any need to implement new regulations to avoid

collisions between marine animals and different ocean vehicles? Some of the marine animals are quite large (in our case we are working with the data of leatherback turtles, each of which generally weighs 200 pounds or more). Are there precautions that should be taken when deploying fishing apparatus?

6. How can nearly extinct marine species, which might be very important for our ecological system, be rescued by observing the lifestyles of their predators?

Besides these specific questions, other unknown factors might be discovered during the research on the above queries. So marine scientists, engineers and biologists attempt to acquire and analyze oceanographic and marine animal data in order to comprehend as well as predict different marine animals' tracks.

1.2   Contributions and Outline

The main contributions of this thesis are to propose and implement novel approaches for visualizing geospatial uncertainty in marine animal tracks. Visualization systems such as Integrated Data Viewer (IDV) [42, 21] or other existing systems such as MAMVIS [12], STAT [8], GeoZui3D [3], GeoZui4D [53] do not offer visualization capabilities specifically designed for our purposes. The data structures of systems such as Visualization for Algorithm Development [20] and IDV are strict and have less flexibility to incorporate different graphical attributes (such as width of line/trajectory, blurring effects, and transparency), which are central to how we visualize our data. Our system has therefore been designed to overcome these problems, as we have developed an application designed specifically for visual analysis of marine animal tracks.

This thesis is divided into five chapters. After Chapter 1, which introduces the problems, and the contributions, the thesis proceeds as follows:

Chapter 2 discusses the broader aspects of visualizing uncertainty from the modeling and visualization perspectives. From a visualization perspective, we discuss existing visualization systems such as MAMVIS (MAMmal VISualization System) [12], STAT (Satellite Tracking and Analysis Tool) [8], GeoZui3D [3], GeoZui4D [53] and methods that researchers have used for tracking and exploring the behavior of different marine animals. From a modeling perspective, we discuss how researchers have analyzed marine animal data using methods such as state-space models (SSMs) and the Bayesian analysis of Markov Chain Monte Carlo (MCMC) methods.

Chapter 3 discusses our novel approach to representing uncertainty of marine animal data. Here we describe state-space model briefly. We discuss our data collection, the source of uncertainty and the implementation of our MarineVis system. Techniques such as how we render and animate our data, and how NASA (National Aeronautics and Space Administration) World Wind terrain is used, are described. We provide the flow diagram and describe the three layers which have been implemented. We also discuss the structure of our visualization system.

Chapter 4 discusses results and evaluation. In this section, we visualize estimated tracks which we obtain using the state-space model. Here we compare our MarineVis system with existing visualization systems and other modeling approaches as well.

Chapter 5 discusses conclusions and future work. For example, in the future, we plan to incorporate different types of NetCDF (Network Common Data Form) layers and graphing methods into our MarineVis system.

Chapter 2

Prior Work / Literature Review

In this chapter, first we attempt to briefly define 'uncertainty' and then discuss the modeling and visualization perspectives of uncertainty regarding marine animals.

2.1   What is Uncertainty?

The National Center for Geographic Information and Analysis (NCGIA) initiative on Visualization of Spatial Data Quality [4] and the National Institute of Standards and Technology's (NIST) Report on Uncertainty [47] defined data uncertainty very broadly to include concepts such as statistical variation, error and a minimum-maximum range. According to the report, uncertainty measurement generally consists of several components, which are regarded as random variables that can be grouped into two categories: the first one is evaluated by statistical methods and the second one is evaluated by other means (such as assigning a probability distribution). Goodchild et al. [15] defined error as the discrepancy between a given value and its true value. Goovaerts [16] described a geo-statistical model of spatial uncertainty. This includes the quantification of precision of interpolated estimates as well as the use of Monte Carlo simulation to describe several possible maps which include a range of plausible spatial outcomes given some observed data. Draper [9] frames the definition in the context of unknown quantities y inferred or predicted on the basis of known quantities x. The model m formalizes assumptions about how x and y are related.

2.2 Modeling Perspective

In this section, we discuss some of the papers that are focused on the modeling of the uncertainty in marine animal data.

2.2.1 Meta-analysis of Animal Movement Using State-space Model

Jonsen et al. [23] stated that using a Bayesian approach and Markov Chain Monte Carlo (MCMC) methods, it is possible to significantly improve the quality of animal tracking data with error measurements and to further improve the quality of data for analysis with the use of state-space models (SSMs). Animals interact with their environment in complex patterns and understanding these patterns for movement behavior [26, 34], and meta-population ecology [36, 19] are important issues in ecology. After an extensive review of existing literature on this subject, Jonsen et al. [23] believe that SSMs are the most appropriate for analyzing animal tracking data [2, 35, 44].

Most of the current methods for analyzing animal movement data, aside from SSMs, use vast simplifications and assumptions about the animals' movement as these relate to their environment and internal states [25]. With the help of satellite, archival and harmonic radar tags, marine and terrestrial animal movement data are collected over large distance [5, 6]. With the existing non-SSM statistical methods, animal behavior is typically classified under the assumption that behaviors are binary (either migrating or foraging), while the SSM method is able to provide a probability for each behavior being estimated from the data. After defining Bayesian equations to compute an estimated path, Jonsen et al. [23] provided a demonstration where they simulated the pathway of an animal with known parameters which they referred as "true pathway" (figure 2.1A, black lines). In real applications, this path is unobservable. Figure 2.1B shows, the "observed pathway" (white lines) which contains

measurement error. Then SSMs were fitted to the data and an "estimated pathway" was produced (figure 2.1C and figure 2.1D black lines) using a maximum likelihood criterion. It was also shown that this technique could be used simultaneously on multiple animal trajectories to better improve estimations of trajectories and meta information. At the end of the paper [23], source code is given which was used to calculate the estimated path as described in this thesis.

2.2.2 Robust State-space Modeling of Animal Movement Data

Jonsen et al. [22] proposed a framework to allow biologists to analyze migratory and foraging phases and time readings of marine animals' tracks. To analyze movement, a correlated random walk (CRW) [48, 34] method was used. Among other things, the CRW method allows behavioral measurements, such as the distinction between migrating and foraging behavior (figure 2.2 and figure 2.3). Because the data used was not always recorded with the same time interval in-between readings, the assumption was made that the animals always move in a straight line between readings.

Using this method, it is also possible to combine datasets recorded at varying intervals. This is a better approach than error handling techniques, which in some cases remove 90 percent or more data [52]. To demonstrate these methods, Jonsen et al. [22] analyzed three data sets (juvenile hooded seal, male gray seal and female gray seal) and they were able to use the output statistics to make distinctions in the habits of these different seals. Some of the observations in these datasets were extreme and dubious in nature (figure 2.2). Figure 2.3 shows a combination of both complex behaviors and numerous extreme observations some of which are on land. Jonsen et al. [22] admitted that they would need to better understand the error distributions since error probabilities are given from the Argos satellite tracking system. They also stated that in the future, they would adapt their model to include existing facts on
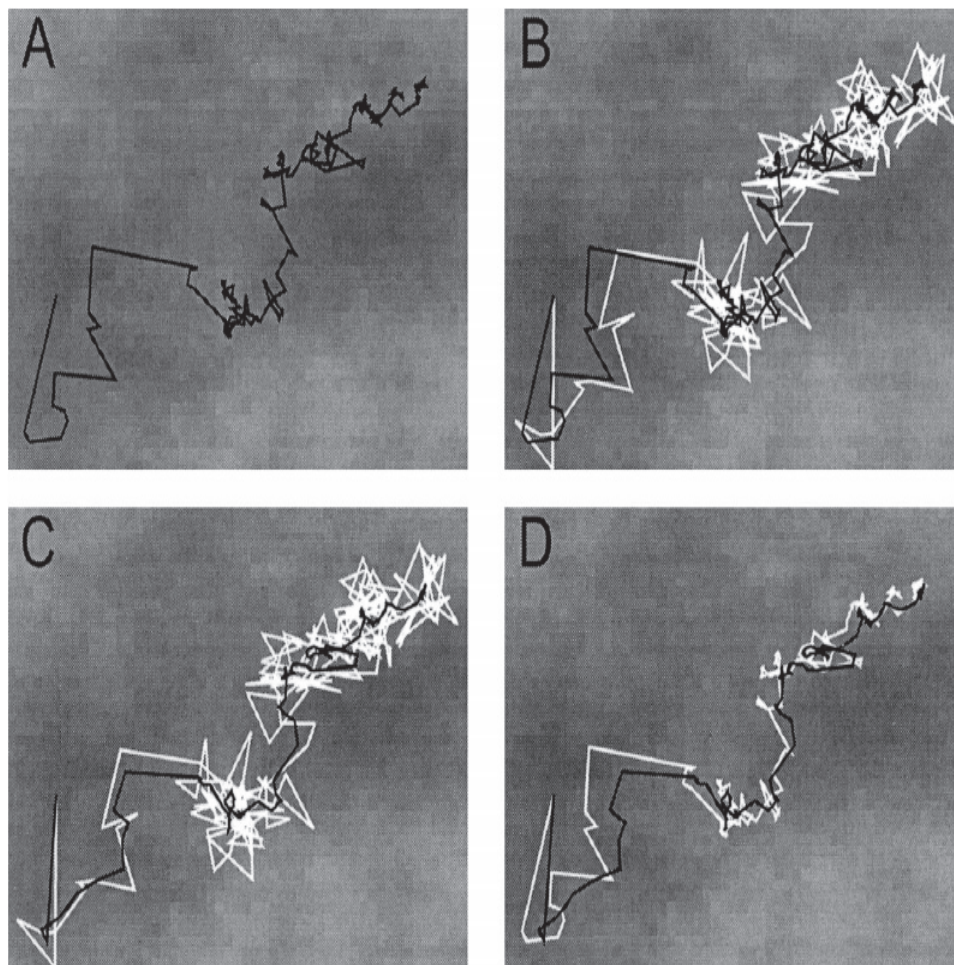
Figure 2.1: Graphical results of fitting a state-space model: This figure shows the results of fitting a state-space model to simulate movement data of a marine turtle. Here (a) depicts the "true pathway"' overlaid on the sea surface temperature, (b) represents the "true pathway" (black) and the "observed pathway" (white) for comparison, (c) represents the plots of the "estimated pathway"' (black) and "observed pathway" overlaid, and (d) represents the plots of the "estimated pathway" (black) and the "true pathway" (white). (Figure is taken from [23]).
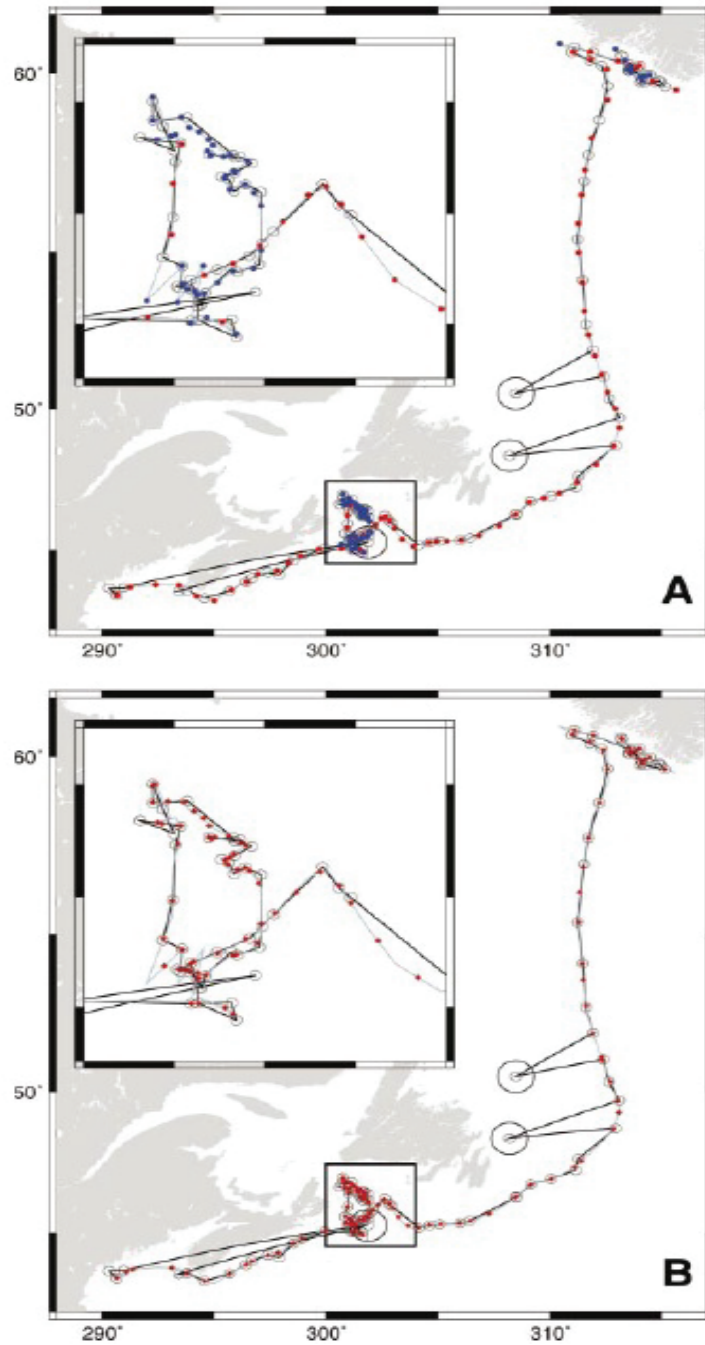
Figure 2.2: Graphical results of fitting a robust MCMC state-space model: (a) This figure shows the plot of track data with observed locations as open circles and state estimates from the DCRWS (first-difference correlated random walk switching) model as red and blue filled circles. Red circles represent migrating and blue circles represent foraging behavior. (b) This figure shows the plot of seal track data with observed locations as open circles and interpolated locations as red diamonds. (Figure is taken from [22]).

the animals and their environment in order to better handle error readings and to provide more accurate behavioral data.

2.2.3 Robust Hierarchical State-space Models Reveal Diel Variation in Travel Rates of Migrating Leatherback Turtles

Jonsen et al. [24] proposed hierarchical Bayesian methods to improve path estimations when using multiple animal trajectories instead of one. They demonstrated this by using this method with state-space models to conduct research on the leatherback turtle's daily movement habits (figure 2.4). They looked at the difference in travel rates between day and night time periods, sex, age, migration stage (north to south or south to north) and whether or not the animal was traveling close to breeding season. A function to calculate the mean travel rate for day/night was defined which also takes into account the fact that day/night depends on the location. They defined a better error distribution than the method provided by Jonsen et al. [22] that fit the real-world satellite recorded data. This error distribution was used by the Bayesian method to improve the estimation of turtle tracks. Jonsen et al. [24] described how they combined these equations and methods to use in their hierarchical Bayesian method.

2.2.4 State-space Models of Individual Animal Movement

Patterson et al. [39] stated that movement patterns of an individual animal and of the population, in general, are important to understand different problems of animals' lives. They also believe that state-space models are the best choice for analyzing tracking data, as SSMs can be designed to deal with problems in the data effectively while minimizing data loss. Movement data is described as a series of positions associated with the times when they are recorded. Patterson et al. [39] noted that from movement data, it is easy to collect other data such as speed, heading, and turn
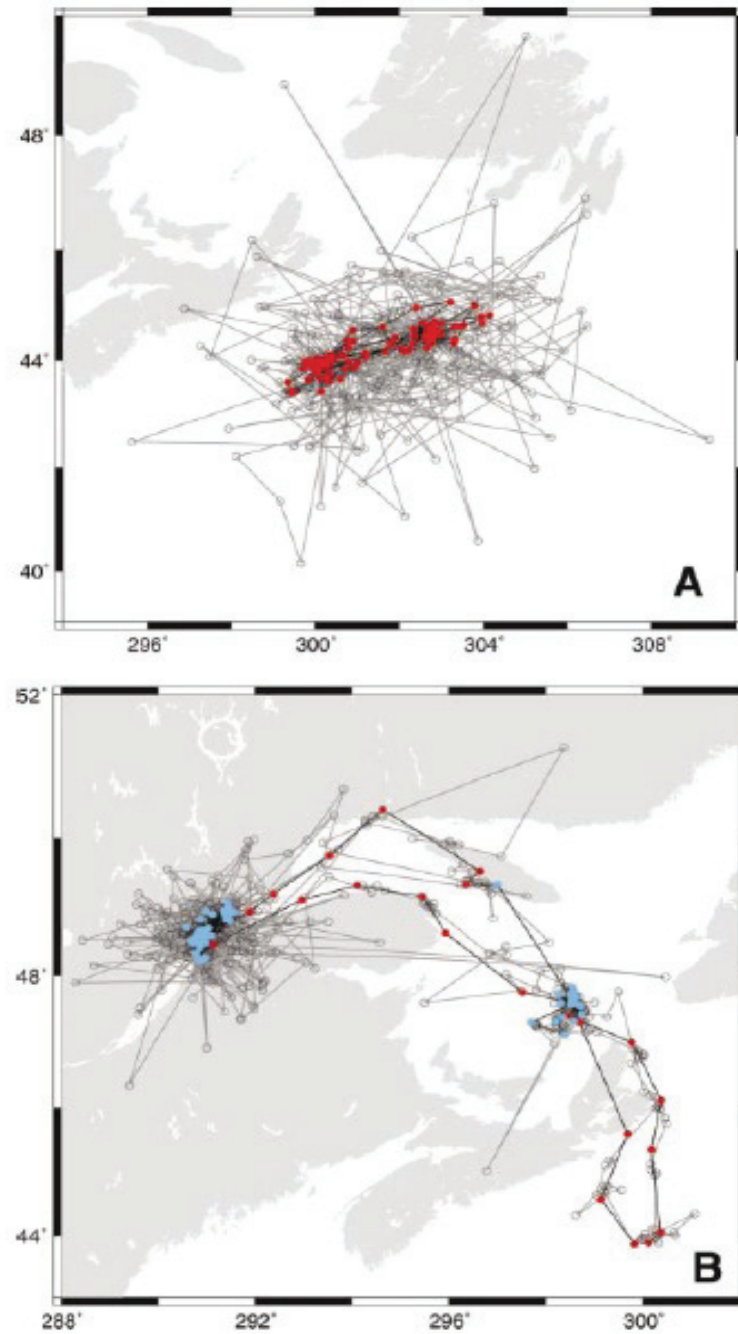
Figure 2.3: Comparison of parameter estimates: (a) This figure shows the plot of male gray seal track data, with state estimates from the DCRWS (first-difference correlated random walk switching) model overlaid in red. This dataset shows numerous dubious observations, some more extreme and others less, and (b) This figure shows the plot of female gray seal track data, observed locations with open circle and interpolated locations with red diamonds. This data set shows a combination of both complex behaviors and numerous extreme observations some of which are on land. (Figure is taken from [22]).
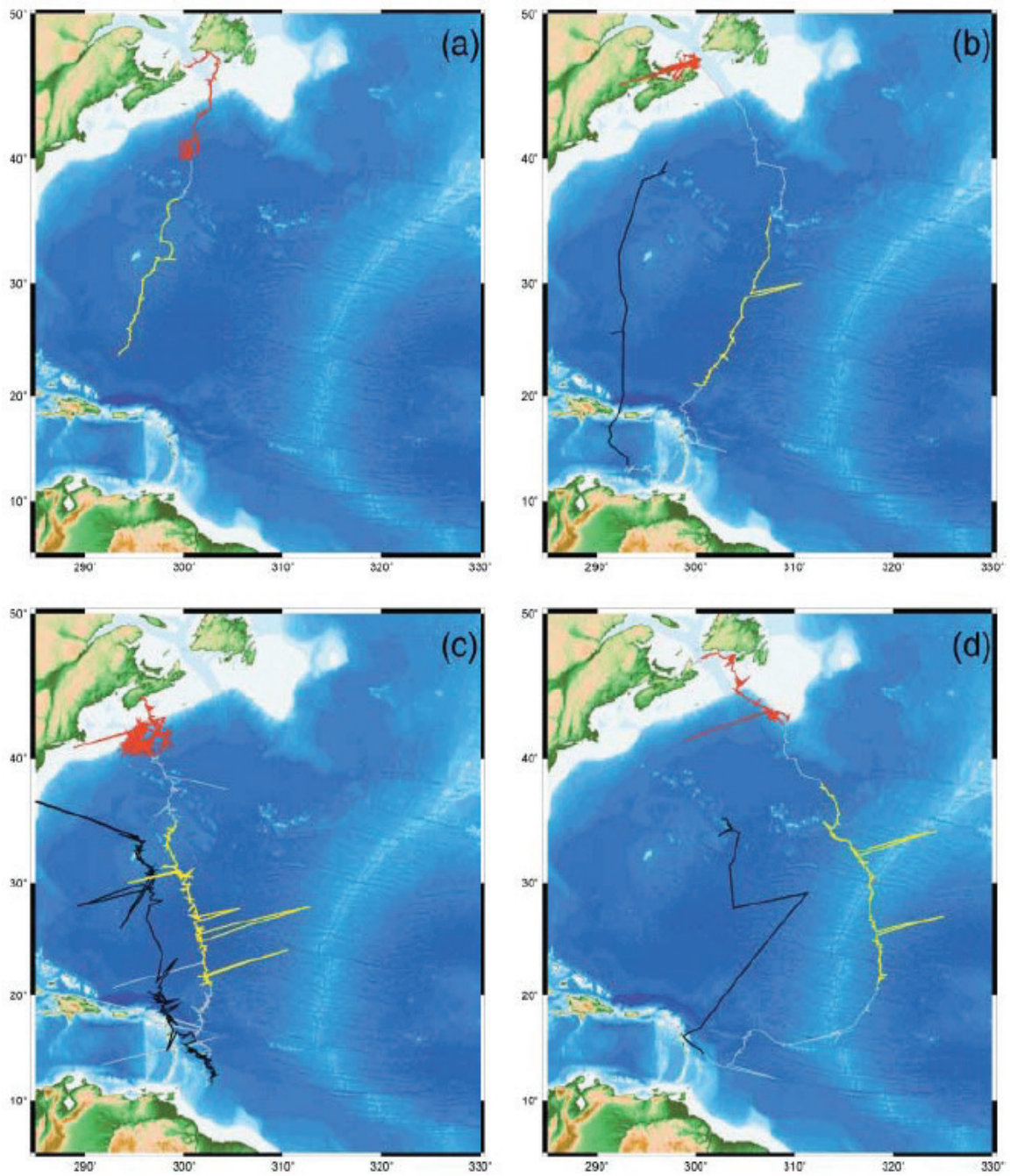
Figure 2.4: Illustration of leatherback turtle pathways: The segments of the pathways used in this figure correspond to northern foraging (red), southward migration (yellow), northward migration (blue), and the rest of the pathways are displayed in white in panels [a-d]. (Figure is taken from [24]).

direction. The data sets are collected using electronic tracking devices of the animal and from 'mark release recapture' (MRR) of the animal. MRR position readings are generally more accurate, however catch and release can disrupt the animal's natural habit and involve much more processing than an automated electronic tracking system. Patterson et al. [39] simulated the different pathways (including data from heterogeneous environment) after applying SSMs on them. In figure 2.5, a hypothetical path is simulated where noisy telemetry data were added. It shows that SSMs can use the characteristics of the track such as speed and turning angle to calculate the probability of the animal being in a foraging mode (figure 2.5). However visualizing in this approach would not work well for a path which is self-crossing. Though it visualizes the uncertainty, it occludes much of the underlying terrain.

The current techniques for error management make simplified assumptions about the way the animal moves, which can potentially lead to erroneous observations about the data. For SSMs, it is necessary to adapt parameters to fit the data and the problem. Because solving an SSM requires an estimative process (section 3.1), Patterson et al. [39] listed two methods to do this; one is Bayesian Monte Carlo and the other is Maximum Likelihood Method (MLM). Methods which are used in MCMC [17] and PF (particle filtering) [1, 40], iterate over randomly generated possibilities until an acceptable solution is found. Patterson et al. [39] stressed that SSMs are more reliable for producing useful statistics on animal movement than any type of custom model.
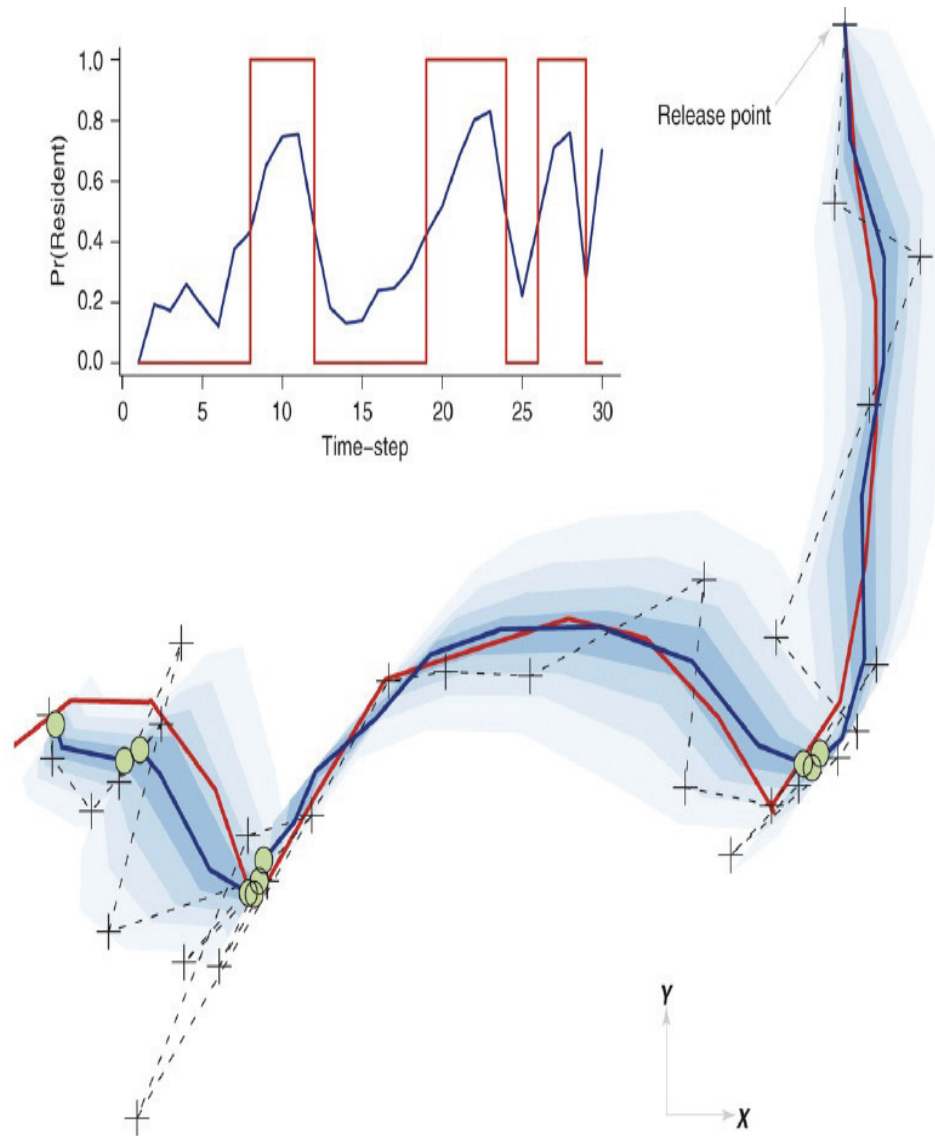
Figure 2.5: Simultaneous inferences of location and behavior: This figure depicts the simultaneous inferences of location and behavior from noisy telemetry data. Black crosses denote noisy observations. Uncertainty in the path is quantified using dark to light blue shading which represents twenty to eighty percent credibility interval. Here the red line represents the hypothetical movement path. Green dots represent the location estimates. (Inset) Two types of behavioral states such as transient and resident states are described. The blue line gives the estimated probability of the animal being in resident mode and the square red line is the true mode. (Figure is taken from [39]).

2.3   Visualization Perspective

In this section, we discuss four papers regarding marine animal tracking and uncertainty from a visualization perspective.

2.3.1   Visualizing the Underwater Behavior of Humpback Whales

Ware et al. [53] investigated ways to track and visualize the underwater movement and behavior of whales. For tracking whale movement, a recording device called digital recording acoustic tag (DTAG) was attached to a whale using suction cups. DTAG is a recording device with accelerometers to collect the gravity vector's direction, axis magnetometers to measure the direction of the earth's magnetic field, a pressure sensor to obtain depth information and a hydrophone to record sound. For visualizing and analyzing the geospatial data collected using DTAG, GeoZui4D was used.

GeoZui4D provides features to visualize a whale's underwater behavior. The capability to play back recorded sound with various other time-varying objects in the scene (for example boats) helps scientists to study marine animals in a unique way as the whales forage. In GeoZui4D, space-time notes are implemented to support the ethological analysis of whale behavior. These space-time notes include variables to represent the space, view direction relative to the look-at point, view scale, point in time at which the note was created, playback rate, name for the note (entered by the user), and typed-in textual information. GeoZui4D also allows to save the space-time notes in a simple XML tagged format. Using track plots, researchers have been able to show various foraging patterns of the whales, for example straight lines with side rolls, repeated side-roll loops with inversion, tightly clustered swimming patterns with side rolls and bubble-net production (figure 2.6).
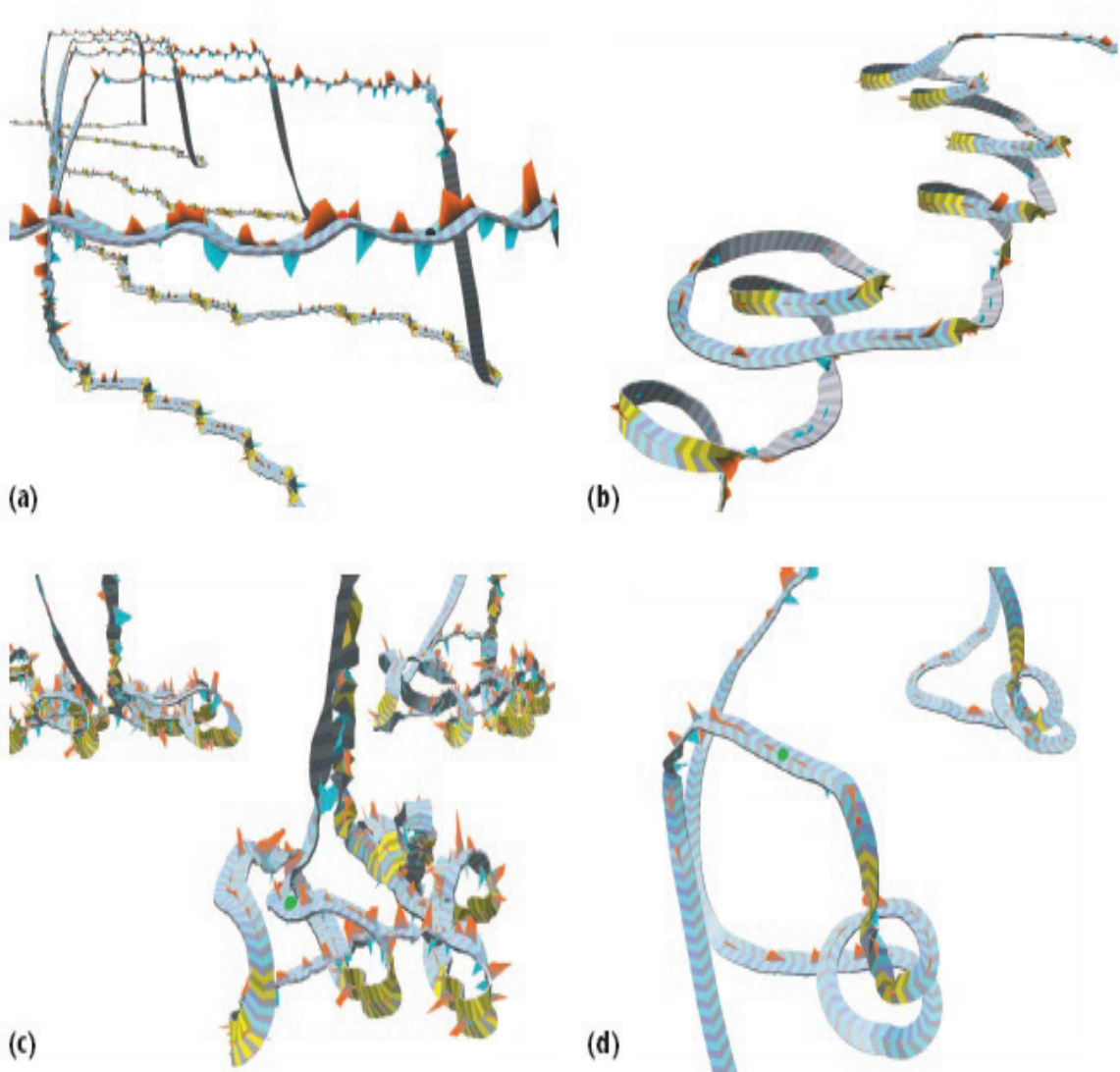
Figure 2.6: Foraging pattern of whales: This figure shows four examples of foraging patterns: (a) Straight lines with side rolls, (b) repeated side-roll loops with inversion, (c) Tightly clustered swimming patterns with side rolls, and (d) bubble-net production. Here sawtooth patterns (red above, blue-green below the dorsal and ventral sides of the ribbon) reveal whale fluke strokes and its amplitude reveals the amount of angular acceleration at any given time. (Figure is taken from [53]).

2.3.2 MAMVIS: A Marine Mammal Behavior Visualization System

Fedak et al. [12] demonstrated the utility of the marine MAMmal VISualization system (MAMVIS) by a case study of the foraging ecology of elephant seals in the South Atlantic. This visualization system provides biologists with the simultaneous display and exploration of oceanographic and environmental data.

MAMVIS displays time-indexed three-dimensional animal behavior in a 3D terrain model that may be overlaid with oceanographic data (figure 2.7). The resulting visualization may be interactively explored by changing the viewpoint, camera lens angle, pan and zoom, and through animation. MAMVIS has the capability of reconstructing track and dive behavior from the raw satellite relay data loggers (SRDL) data. After that, the data is stored in a database and can be retrieved using SQL queries. The locations that form the track data are subjected to error and are filtered by an algorithm described by McConnell et al. [31]. Several modules are used to perform different tasks of MAMVIS. The track limits module determines the geographical range of the selected tracks that are used by the other modules to set the extent of the geographical window to be displayed. The animated time module displays an animated time series of track data and controls the animation. The draw coded tracks module controls the display of track and dive data. The shoreline module draws the coastline within the geographic window defined by track limits. The image information created by the modules draw coded tracks; sea surface temperature, shoreline, bathymetry and grid are displayed by the AVS module geometry viewer.

Fedak et al. [12] also illustrated the use of MAMVIS by tracking the movements and foraging behavior of southern elephant seals (figure 2.8, and figure 2.9). MAMVIS allows users to see the tracks of seals over the sub-sea topography with dives shown as
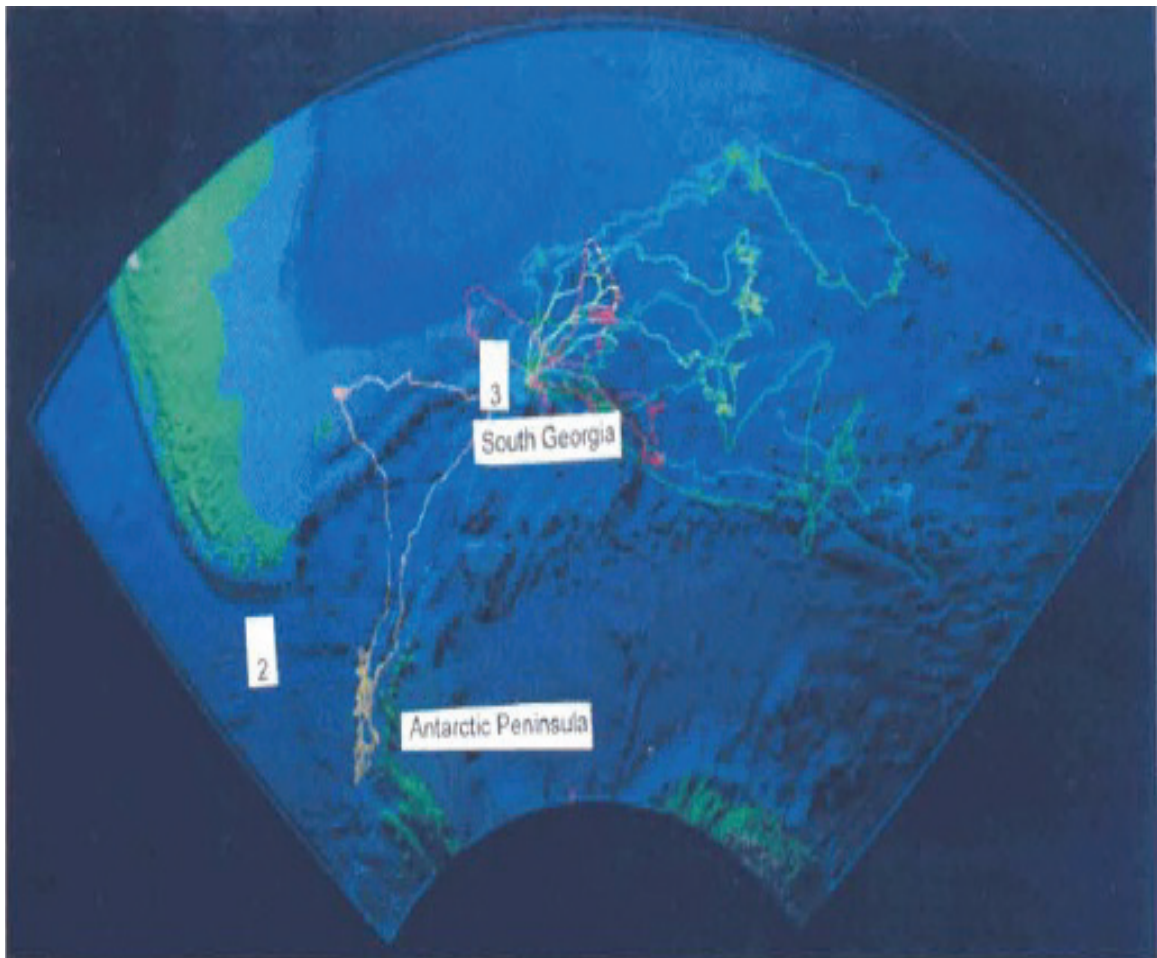
Figure 2.7: Tracks of seals on sea bed of topography: This figure shows the tracks of twelve southern elephant seals tracked from South Georgia overlaid on sea bed topography. (Figure is taken from [12]).
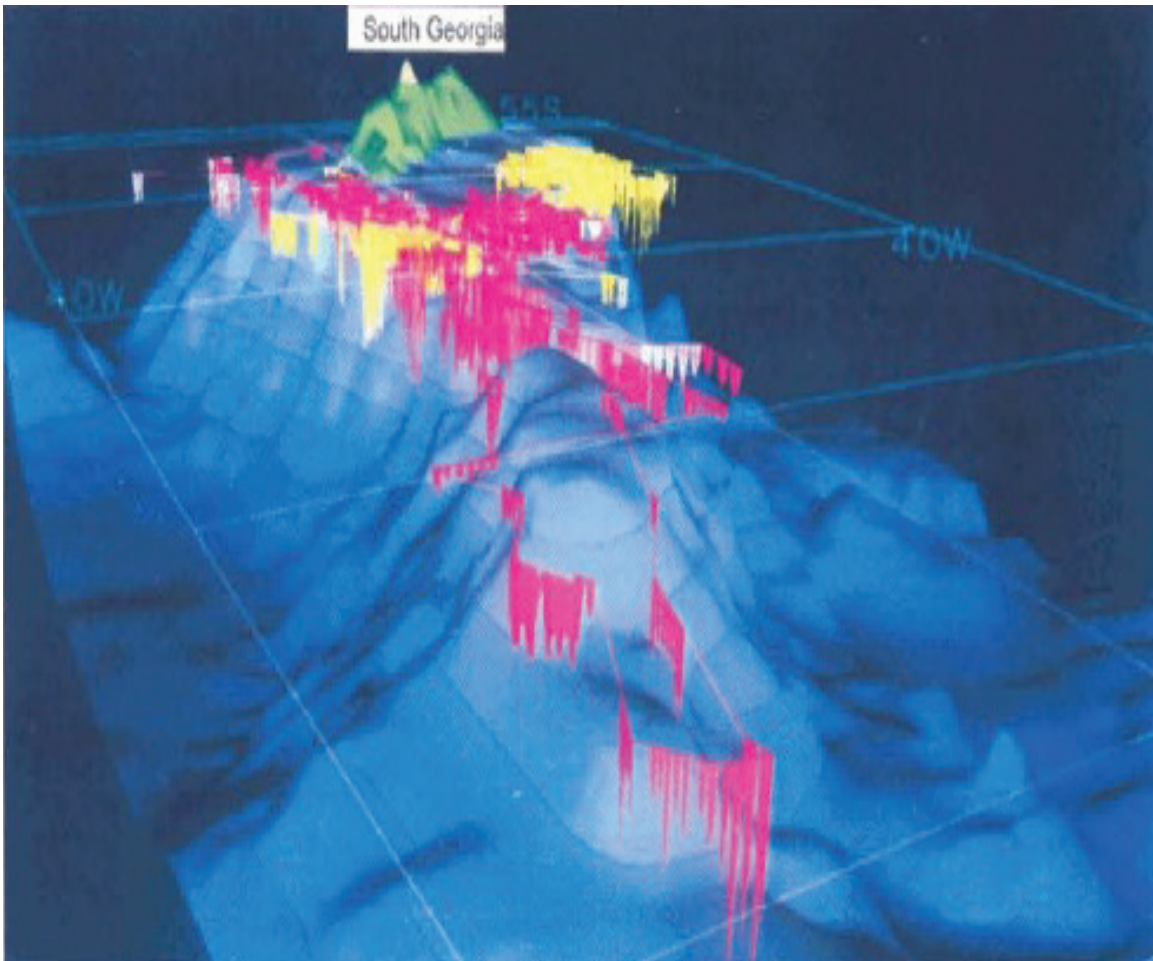
Figure 2.8: Tracks and dive profiles of two elephant seals: In this figure, the vertical shapes show the depth profile of individual dives. Here seabed, dives and depth both are exaggerated by a factor of twenty. (Figure is taken from [12]).

lines or shapes below the track. Using these tracks, it is possible to identify various movement patterns of the seals. MAMVIS also allows users to easily examine the behavior of these animals and provides insight into the biology of elephant seals in the open sea.
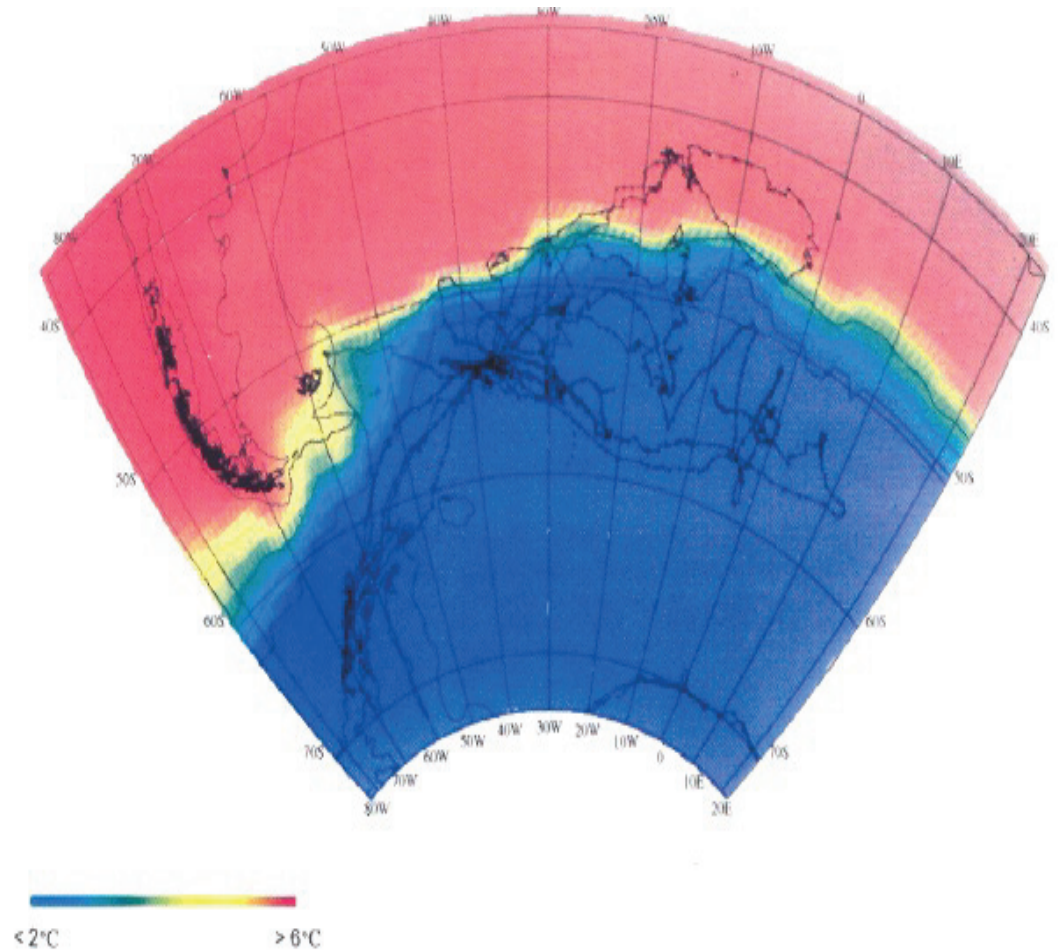
Figure 2.9: Seal tracks overlaid with temperature: This figure shows the tracks of the southern elephant seals overlaid with sea surface temperature. (Figure is taken from [12]).

2.3.3 A System for Visualizing Time Varying Oceanographic 3D Data

GeoZui3D is a 3D geo-referenced data visualization system with a zooming user interface that allows users to rapidly zoom in and out with respect to points of interest. The zoom in/out facility is available for both time and space; zoom into time refers to slowing down of the dynamic process in more detail and zoom out in time means speeding it up. To show the views of 3D data that changes through time, GeoZui3D

continuously updates the view so that all viewing operations are respect to the reference frame of the moving object of interest.

Arsenault et al. [3] enhanced the GeoZui3D system to handle multiple time varying data sets from heterogeneous sources to understand dynamic system behaviors in various processes such as biological processes, physical oceanographic processes, the motion of instrument platforms (such as ships, ROVs and AUVs), and the interactions between these data. Their research addressed the problem of creating a user interface supporting rapid spatial and temporal navigation through such a 4D data set.

Enhanced GeoZui3D can be used for visualizing flow fields, understanding whale behavior, planning a voyage with a tide-aware 3D chart, and showing real-time data from an oceanographic vehicle. To visualize flow, GeoZui3D is enhanced to provide the easy interfacing with external flow models such as CBOFS [18]. It also allows users to interactively drop floaters (objects which can flow in the simulation model) into the flow model and watch their flow path. The NetCDF file format is used to support real time tracing and animation. GeoZui3D can also show larger numbers of tracers if their paths have been pre-computed. It has the capability to place the still images in correct geo-referenced locations, and it can also play back sound recorded from the whale tag (figure 2.10). The zoom in/out feature enables a researcher to rapidly switch between whale-centered and world referenced views. Using both frames of reference, the time controller could be used to move backwards and forwards in time and to speed up and slow down the animation in order to examine behaviors on different time scales. Arsenault et al. [3] also developed GeoNav3D which is a variant of GeoZui3D for navigation decision support. GeoNav3D enables users to see tidal states along a path by displaying the anticipated tides at the estimated time of
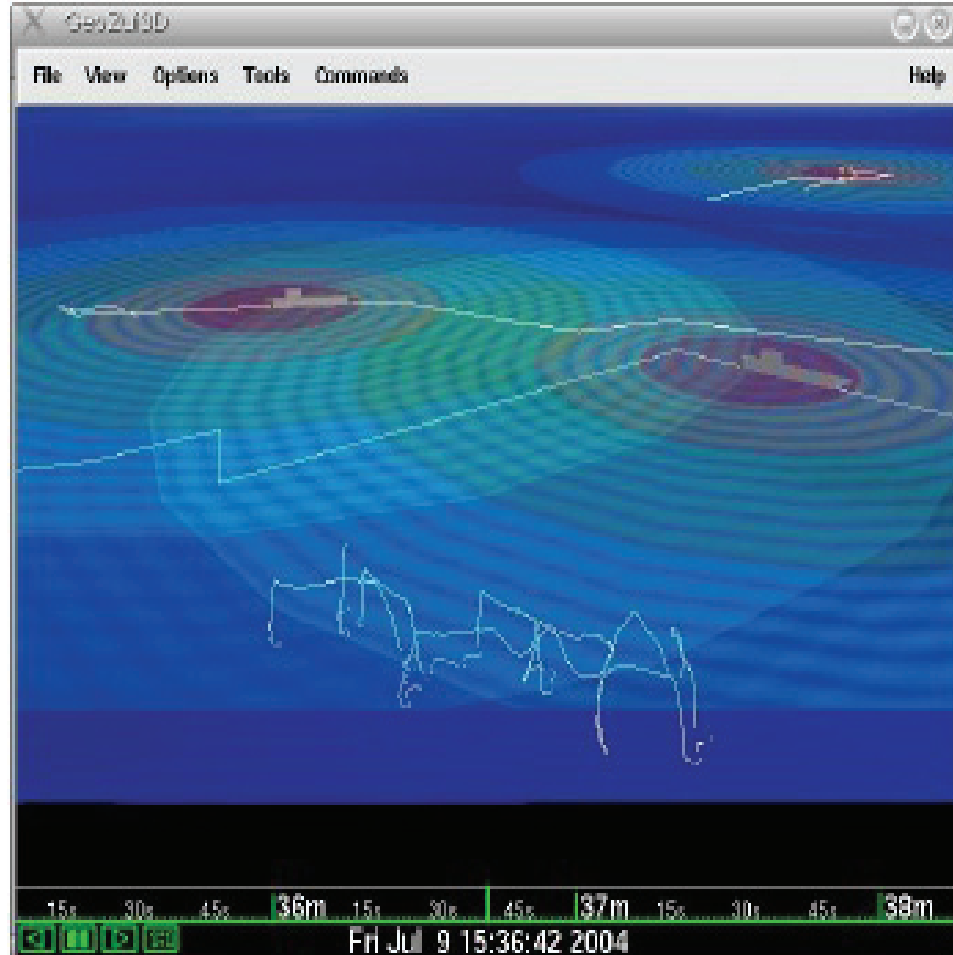
Figure 2.10: Sound in GeoZui3D: This figures shows an overview of the sound fields from fishing vessels in the vicinity of the whale. (Figure is taken from [3]).

arrival at each point along the path, within a corridor surrounding the path. Another contribution of Arsenault et al. [3], is to provide real time display of various scientific and navigation sensor data obtained from oceanographic research.

2.3.4   The Satellite Tracking and Analysis Tool (STAT)

The Satellite Tracking and Analysis Tool (STAT) [8] system contains a set of standardized tools and techniques for data management, analysis, and integration with environmental data. Since the late 1980s, the Argos system has been the predominant

satellite-based system for tracking wildlife. But with the increase in the number of tracked animals using the Argos system, the complexity and scope of the animal studies increased and researchers are increasingly facing the need to effectively manage these datasets. To improve data management, analysis, comparison of Argos data, the STAT system has been developed.

The STAT system is an entirely web-based and platform-independent system. It uses Perl scripts to execute various functions, Telnet to retrieve data from Argos and FTP to obtain environmental data from data servers such as NOAA (National Oceanic and Atmospheric Administration), and NASA. STAT logs into the Argos computer network each day and downloads all available locations and associated data for each user. These data are parsed and stored in a relational database in STAT. It allows the data owner to download all the data to a flat-file format, commonly CSV, which can be directly imported into most spreadsheet, statistics and GIS software packages. It is also possible to generate maps using the data layers.

The STAT system allows data filtering and editing facilities. Its Maptool facility allows users to generate publication quality maps within the STAT online framework. Integration of a variety of data layers with animal tracking is also an important feature of STAT. A variety of environmental data layers (figure 2.11) such as sea surface temperature, sea surface height, ocean currents and chlorophyll can be sampled for all locations in the STAT database and can be downloaded and incorporated into tracking maps and animations.
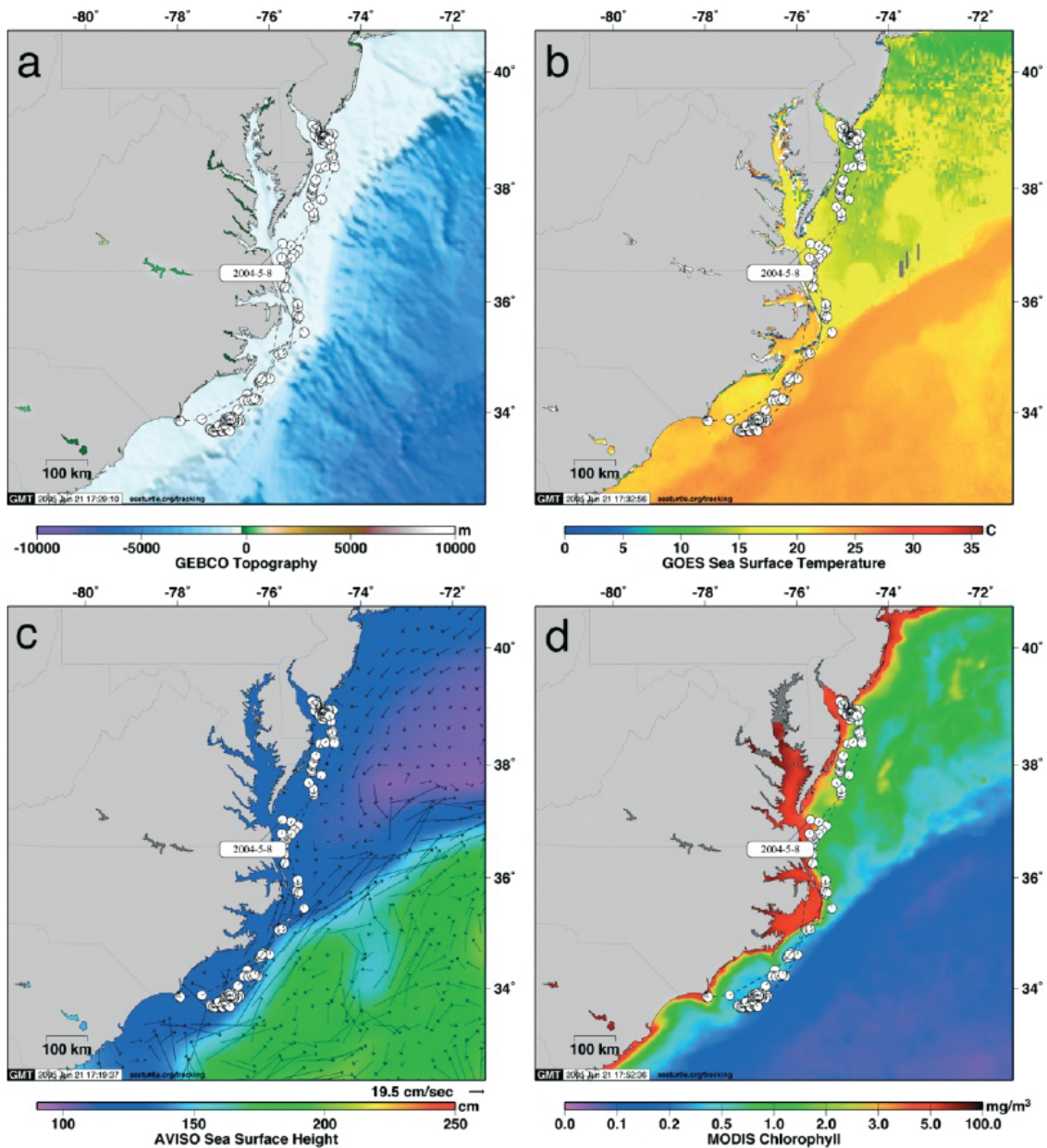
Figure 2.11: Output from the STAT package for a loggerhead turtle: This figure shows the output of STAT package for a loggerhead turtle tracked from Bald Head Island, North Carolina to Delaware Bay. The above figure shows (a) GEBCO (General Bathymetric Chart of the Oceans) bathymetry, (b) NOAA (National Oceanic and Atmospheric Administration) GOES (Geostationary Operational Environmental Satellites) sea surface temperature, (c) Geotropic currents and sea surface height, and (d) Chlorophyll layers obtained using MODIS satellite-based sensors [41]. (Figure is taken from [8]).

2.4 General Ways to Visualize Uncertainty

Generally the usage of glyphs, graphical objects, animations, volume rendering and iso-surface rendering are the standard techniques for visualizing uncertainty.

A glyph is a geometrically plotted symbol that encodes values from data. Using different kinds of shape, color, and transparency glyphs encode information. Magnitude and directional information can be visualized using arrow glyphs. Radiosity [37], vector fields [55], surface interpolation [27], flow visualization [27] and key-framed animation [54] can be visualized using glyphs. One of the main reasons for using glyphs is to represent the sampling frequency/location and placement orientation [49]. Wittenbrink [54] proposed a method of using uncertainty glyphs that are scaled to magnitude. Lower glyphs are used to visualize low uncertainty areas and higher glyphs to visualize high uncertainty areas. Wittenbrink et al. [55] also used glyphs for vector fields. Another technique used by MacEachren [30], employs several glyphs where the areas with high uncertainty are visualized with more clear glyphs, and the areas with low uncertainty with less clear glyphs. The problem with this technique is that only two predefined types of glyphs are used, i.e., clear and unclear and hence it is unable to visualize different degrees of uncertainty using only these two symbols. Ebert et al. [11, 10] and Shaw et al. [43] used procedurally generated glyphs to visualize multi-dimensional data. They developed a system named Stereoscopic Field Analyzer (SFA) which supports the recognition of the changes of uncertainty in the data based on these glyphs. These types of glyphs convey the uncertainty in both orientation and magnitude. Lodha et al. [27] combined their techniques to create UFlow, a method to view the uncertainty in fluid flow using different numerical integration algorithms and different time steps. They used vector based 3D glyphs that ranged from path envelopes and ribbons to batons and barbells to visualize fluid flow

differences. Lopes and Brodlie [28] also looked at this problem. They used strips and tubes to visualize differences. Pang et al. [38] used different kinds of techniques to visualize uncertainties which range from modifying or adding geometry to altering lighting attributes and the use of textures. One of their techniques is to use a blurring effect by which users can easily perceive subtle changes of uncertainty. One common problem of using glyph-based 3D visualization is when the glyphs overlap, the visualization becomes cluttered.

Researchers have used animation and sound to visualize uncertainty as well. Monmonier [33] created and played back sequenced images stored as graphics scripts. MacEachren [30] used sequential presentation and interactive animation to visualize uncertainty. Another approach proposed by Monmonier [32] where real-time animation of random dots was used to show uncertainty in spatial information. Similarly, Gershon [13] used animation loops to examine the role of motion detection. In our system, we have combined different techniques such as glyph/geometry, and animation to visualize the uncertainty of the turtle tracks.

Chapter 3

The MarineVis System

In this chapter, we describe state-space models, the sources of uncertainty and the implementation of our MarineVis system. Techniques such as how we render and animate our data, and how NASA World Wind terrain is used, are described. We provide a flow diagram and describe the three layers which have been implemented.

3.1 State-space Model

SSMs are time-series models that allow unobservable, true states to be estimated from observed data by incorporating errors arising from imprecise observations. In a state-space model, there are unobserved states of interest that evolve through time, and partial observations of the states are made at successive time-points.

In a Bayesian context, one of the inference objectives for SSMs is generating samples from the posterior distribution for the states and unknown parameters conditional on the entire observation time series. In order to fit marine animal data, Jonsen et al. [23] formulated SSM in the following way:

A marked animal's position is observed at a series of times $t = 1, 2, ..., T$. The locations are observed with error which gives rise to a measurement equation

$$y_t = h_t(\alpha_t, \varepsilon_t), \ t = 1, 2, ..., T \qquad (3.1)$$

where, y_t is the observed location, α_t is the estimated location (the state variable), and ε_t is the measurement error. Unobserved locations are calculated using transition equation

$$\alpha_t = f_t(\alpha_{t-1}, \eta_t; \gamma) \tag{3.2}$$

Here, γ is a vector of parameters describing the movement process variable and η is process noise that is usually assumed to be normally distributed.

The purpose of the SSM is to estimate the unobserved locations, α_t, using the above two equations. If y_t contains all the observations from time 0 through t, then $p_y(y_t|\alpha_t)$ is the density function derived from the measurement equation, and $p_\alpha(\alpha_t|\alpha_{t-1}; \gamma)$ is the density function derived from the transition equation.

State-space models are formulated in two steps [45, 46]:

1. Prediction, where new location is predicted from all the previously observed data, Y_{t-1}. The prediction equation is:

$$p(\alpha_t|Y_{t-1}; \gamma) = \int p_\alpha(\alpha_t|\alpha_{t-1}; \gamma) p(\alpha_{t-1}|Y_{t-1}; \gamma) \, d\alpha_{t-1} \tag{3.3}$$

2. Update, where the new information provided by y is updated using Y_{t-1}. The update equation is

$$p(\alpha_t|Y_t; \gamma) = \frac{p_y(y_t|\alpha_t) p(\alpha_t|Y_{t-1}; \gamma)}{\int p_y(y_t|\alpha_t) p(\alpha_t|Y_{t-1}; \gamma) d\alpha_t} \tag{3.4}$$

Using a Bayesian approach or a frequentist approach, SSMs can be fitted to data. In the Bayesian approach, it is possible to use the Markov Chain Monte Carlo method to solve the integration problem which is used in [23]; hence our input data is generated using this approach. Extensive details of the use of MCMC can be found in [14].

BUGS [29] and WINBUGs software use MCMC techniques to generate samples from posterior distributions.

3.2 Visualization Approach

At the beginning of our research, we started visualizing our data using existing software such as IDV [42]. Due to the rigid structure of VisAd [20] and as IDV [42, 21] is implemented using Visad, it is not possible to include uncertainty information in the tracks in IDV. Besides a blurring effect, the width of the track cannot be changed in IDV.

To overcome these limitations, we have developed an interactive visualization system to visualize the uncertainty in marine animal data generated using MCMC models. In our system, we have applied combinations of techniques to visualize positional and behavioral uncertainty of marine animals. We use the output generated from WinBUGs code supplied by Jonsen et al. [23] as input of MarineVis to visualize uncertainty in turtle tracks, confidence intervals, and uncertainty in marine animals' migrating and foraging behavior.

We design MarineVis as a layer based system, where in each layer, independent visualization techniques can be included. To understand the turtles' geographic location, a background map is needed; hence our first layer is the NASA World Wind terrain. Our second layer is the Confidence Interval layer. For this, we use ellipses to visualize the confidence intervals which we obtain using the SSM method and triangles as wedges to follow the directions. Our third layer visualizes the estimated animal trajectory by using mean and standard deviation (table 3.1).

3.3   Data Collection and Sources of Uncertainty

We have collected our data from the Bedford Oceanographic Institute of Nova Scotia, Canada. These data are mainly turtle track files, which originally are collected using satellite tags, and NetCDF files.

On this data, we apply SSM code provided by Jonsen et al. [23] that generates estimated tracks and behavior of the turtles (table 3.1) which we use as input of our visualization system. In the table 3.1, there are seven columns. The first one is the name of the variables and the subsequent columns are mean, standard deviation, MC error, observed lower percentile being reported (default 2.5 percentile), median, and observed higher percentile being reported (default 97.5 percentile). Here 2.5 percentile and 97.5 percentile are the credible intervals and 'MC error' gives the variability of each estimate due to the simulation and it should be low in order to calculate the parameters with high precision.

NetCDF files can be of different conventions [50] such as CF (Climate and Forecast), COARDS (Cooperative Ocean-Atmosphere Research Data Service), NUWG (NetCDF Users Working Group), and CDC (Common Data Format). Our data is in the "Climate and Forecast (CF)" convention. The CF convention provides additional information with standard variable names (such as "sea_water_potential_temperature", "sea_water_potential_velocity", "sea_water_potential_elevation"). This additional information is very important to comprehend how marine animals interact with the environment and how those factors affect the trajectories of the animals during their migrating and foraging phase.

Node statistics

node	mean	sd	MC error	2.5%	median	97.5%
Sigma[1,1]	0.01786	0.001991	7.272E-5	0.01427	0.01771	0.02216
Sigma[1,2]	-9.499E-5	9.955E-4	2.129E-5	-0.002042	-1.036E-4	0.001859
Sigma[2,1]	-9.499E-5	9.955E-4	2.129E-5	-0.002042	-1.036E-4	0.001859
Sigma[2,2]	0.01119	0.001124	2.767E-5	0.009213	0.01113	0.01361
alpha[1]	0.9038	0.04099	0.00149	0.8121	0.9087	0.9693
alpha[2]	0.07475	0.03997	0.002123	0.01904	0.06685	0.1742
bmode[1]	1.681	0.4659	0.01356	1.0	2.0	2.0
bmode[2]	1.733	0.4425	0.01377	1.0	2.0	2.0
bmode[3]	1.71	0.454	0.01359	1.0	2.0	2.0
bmode[4]	1.164	0.3703	0.007308	1.0	1.0	2.0
bmode[5]	1.038	0.1918	0.003206	1.0	1.0	2.0
bmode[6]	1.0	0.02236	3.514E-4	1.0	1.0	1.0
bmode[7]	1.0	0.02236	3.514E-4	1.0	1.0	1.0
bmode[8]	1.001	0.02738	4.327E-4	1.0	1.0	1.0
bmode[9]	1.001	0.02738	4.295E-4	1.0	1.0	1.0
bmode[10]	1.0	0.0	1.118E-12	1.0	1.0	1.0

A

node	mean	sd	MC error	2.5%	median	97.5%
x[1,1]	-63.99	0.01516	2.711E-4	-64.02	-64.0	-63.96
x[1,2]	44.5	0.01256	1.951E-4	44.48	44.5	44.53
x[2,1]	-63.86	0.006733	1.126E-4	-63.87	-63.86	-63.85
x[2,2]	44.35	0.005276	7.739E-5	44.34	44.35	44.36
x[3,1]	-63.76	0.01225	1.726E-4	-63.79	-63.76	-63.74
x[3,2]	44.44	0.011	1.823E-4	44.42	44.44	44.46
x[4,1]	-63.78	0.009393	1.633E-4	-63.8	-63.78	-63.76
x[4,2]	44.22	0.007109	1.176E-4	44.21	44.22	44.24
x[5,1]	-63.92	0.0437	6.268E-4	-64.01	-63.92	-63.84
x[5,2]	44.09	0.02617	4.704E-4	44.04	44.09	44.15
x[6,1]	-64.23	0.01129	1.865E-4	-64.25	-64.23	-64.21
x[6,2]	44.11	0.007895	1.373E-4	44.1	44.11	44.13
x[7,1]	-64.57	0.008906	1.535E-4	-64.59	-64.57	-64.55
x[7,2]	44.06	0.008988	1.523E-4	44.04	44.06	44.07
x[8,1]	-64.75	0.008458	1.739E-4	-64.77	-64.75	-64.73
x[8,2]	43.85	0.005995	1.055E-4	43.84	43.85	43.86
x[9,1]	-64.8	0.01599	0.001013	-64.87	-64.8	-64.79
x[9,2]	43.68	0.008854	2.503E-4	43.66	43.68	43.69
x[10,1]	-65.06	0.008162	1.276E-4	-65.08	-65.06	-65.05
x[10,2]	43.56	0.003524	5.807E-5	43.55	43.56	43.57

B

Table 3.1: Node statistics of WinBUGs output. This table shows the partial results generated using turtle data and applying SSM code supplied by Jonsen et al. [23]. In table A, 'bmode[1]' refers to the behavioral mode of the turtle and in table B, 'x[1,1]' refers to the longitude of the first node and 'x[1,2]' refers to the latitude of that node.

3.4 Implementation Details

We have implemented an interactive visualization system where by applying a combination of different techniques, we visualize turtle behavior, trajectories and other features. Using MarineVis, marine animal data, their migrating and foraging behavior, uncertainty in their position and in their behavior can be visualized. The concept of MarineVis is similar to the open-layer concept where we add several layers, one on top of another. Currently three layers have been implemented. Our first layer is the NASA World Wind terrain, the second layer is the Confidence Interval layer, and the third layer is the Animal Trajectory layer. Subsequent layers such as Comparison of Confidence Intervals layer and NetCDF layers to be implemented in the future. MarineVis is implemented using Java and Java2D from the ground up without making use of any additional libraries (such as JOGL, Visad). The structure of the MarineVis is outlined in the flow diagram shown in figure 3.1, which is described in the subsequent sections.

3.4.1 NASA World Wind Terrain

MarineVis uses the multi layered, multi-tiled NASA World Wind database. This database is pre-cached so that navigation through the map or different animation states is very fast and efficient. Users are able to select any particular state/node, zoom in/zoom out, translate and re-scale the map. Our first layer is the NASA World Wind terrain. The NASA World Wind map is in equirectangular projection (also called plate carre projection, carte parallelogrammatique projection, geographic projection). This projection maps meridians to equally spaced vertical lines and circles of latitude to evenly spread horizontal straight lines. Here the point (0, 0) is at the center of this resulting projection and the input range is $[-180°, 180°]$ rather than $[0, 360°]$. This projection is very suitable and standard for computer applications
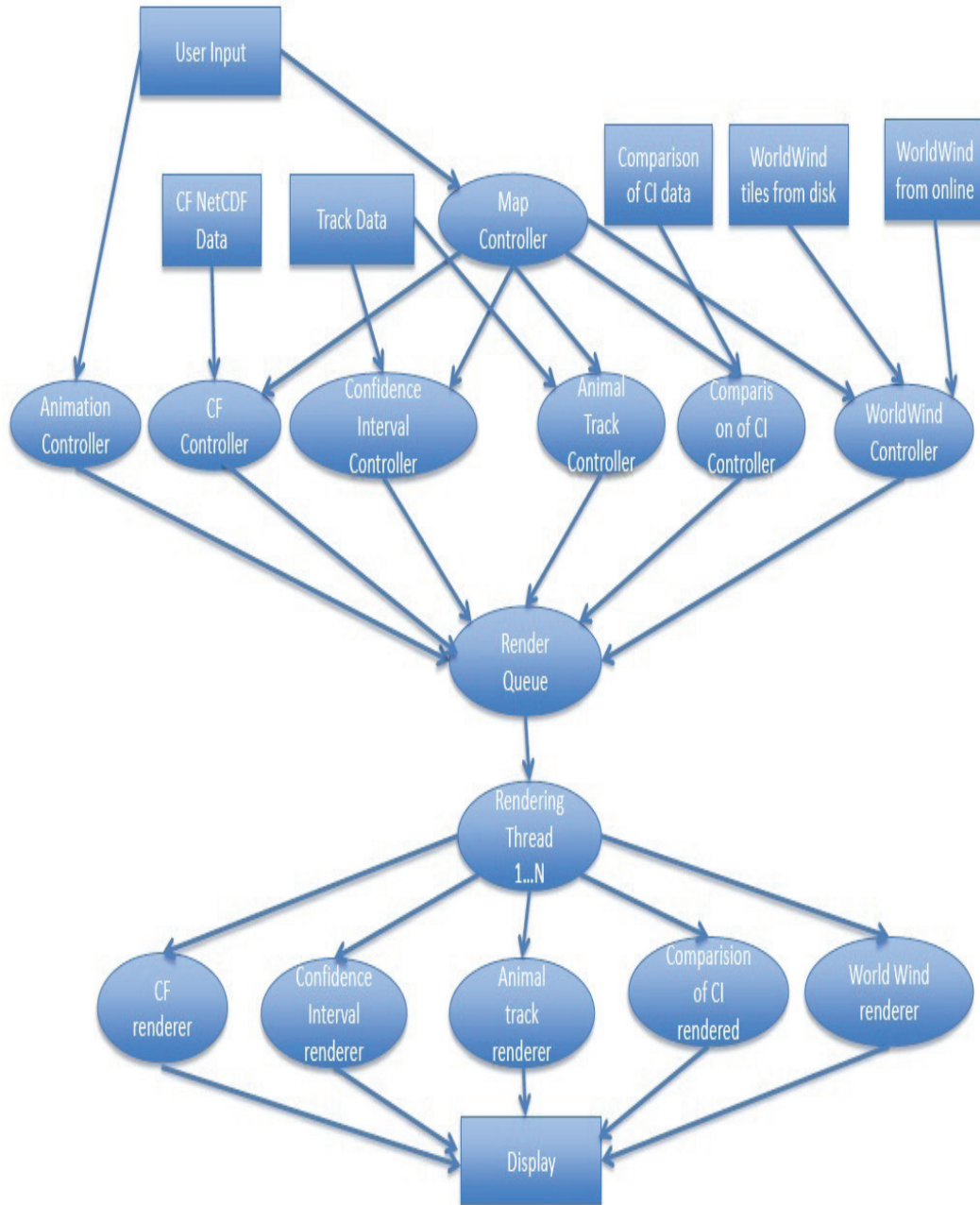
Figure 3.1: Flow diagram of the MarineVis system

particularly because of the simple relationship between the position of an image pixel on the map and its corresponding geographic location on earth.

A tile is a geographical extent in the form of a bounding box where minimum longitude and maximum latitude represent the top left corner and the maximum longitude and minimum latitude represent the bottom right corner of the tile's bitmap. Each tile is an image or rendered map of the ground at a predetermined fixed level of detail.

World Wind uses "Level Zero Tile Size" to determine how large each tile would be in width and height. The level zero tile size (referred as $lzts$) is the distance traveled in degrees from one side of a tile to the opposite side. The function for loading tiles is described in detail in Appendix A.

Tile Rendering Techniques

In our system, (Util.drawTile) is the tile rendering function which is responsible for loading, positioning, translating and rendering the tile into the viewport. The tile rendering techniques is as follows:

1. Calculate the difference between top left corners of the tile and the viewport (in geographical units)

    (a) *loffx = tile.min.longitude - viewport.min.longitude,*

    (b) *loffy = viewport.max.latitude - tile.max.latitude.*

    This difference gives us the rendering position, but in geographical units instead of viewport pixels.

2. Translate the difference calculated in step 1 into viewport pixel units.

    (a) *x = [(loffx / viewport.geo.width) * viewport.pixels.width],*

(b) $y = [(loffy / viewport.geo.height) * viewport.pixels.height]$.

3. Calculate the scale by which to render the tile.

   (a) $scalex = (tile.geo.width / viewport.geo.width) * (viewport.pixels.width / tile.pixels.width)$,

   (b) $scaley = (tile.geo.height / viewport.geo.height) * (viewport.pixels.height / tile.pixels.height)$.

4. Render the tile given the coordinates (x, y), and the scales $(scalex, scaley)$.

3.4.2 Confidence Interval Layer

In this layer, we use ellipses to visualize confidence intervals and triangles to visualize the direction of the tracks. The radii of each ellipse (figure 4.2) are obtained from the 2.5 percentile and 97.5 percentile credible intervals (table 3.1). For each reading, the process is as follows:

1. If the reading is not the first in the range to be rendered, draw the triangle opening into it.

2. Next the aura around the ellipse is rendered.

3. Finally the ellipse is rendered using Java's Graphics2D built in fillOval function. The color of the ellipse becomes yellow if selected otherwise it is white.

Unit conversion of dynamically rendered layers is described in Appendix B, where it is described how we convert geographical data points into viewport pixel points.

3.4.3 Animal Trajectory Layer

In this layer, we render a series of ellipses to form the track. Each ellipse's center is the mean, and its radii are the standard deviations of longitude and latitude which

are obtained using the SSM code provided by Jonsen et al. [23] on turtle data (table 3.1). This layer is rendered incrementally or all at once, depending on the animation position. The animation index is a floating point value (a value of 0 indicates the first point, a value of 1 the second and so on). Intermediate values represent points in between. For example, a value of 1.5 indicates a point half way along the line segment formed by the second and third readings.

The rendering function consists of two nested loops:

1. (Outer) Point level which is responsible for changing the line segment.

2. (Inner) Line level which is responsible for rendering each ellipse along a line segment between readings.

The inner loop renders a number of ellipses and updates the animation index accordingly. A copy of the fractional portion t is updated in parallel; this is kept separately to make it easy to detect when a boundary is crossed (i.e., when we need to exit the inner loop and switch to the next line segment). The (x, y) centre of the ellipse to be drawn, the (r_x, r_y) radii, and the bmode BM are calculated using linear interpolation, with the position index based on the fractional part of the animation index.

Animation System

The animation system is contained within a custom control, which consists of:

1. A thread to keep track of time.

2. A slider control to show time and allow the change of time.

3. Play buttons (1x, 2x, 4x) to start/stop the animation and control the speed.

There is a single thread for each animation control; this thread is always running. Every time the slider position changes, a message is sent which initiates the animation control, via the 'updateAnim' function. It is up to the specific layer to decide what to do with this information. Generally, the overlay's tile is re-rendered according to the changes in animation.

Code Optimization

Without optimization, each tile would be rendered in a separate thread without communicating with other layers. This would cause the tile loading and rendering process to become slow and stall for the user. When we added more layers, this compounded the problem which made our system extremely slow. To solve this problem, a centralized rendering queue has been created. From the point of view of a layer, when it queues the tile:

1. It requests a single instance of the RenderQueue object.

2. Uses the "push" method to queue the tile.

From the RenderQueue perspective, when the single instance of this class is first created, it creates a number of threads (such as a worker thread where the default number is 1) for the process of rendering and loading jobs. Each worker thread for the Render Queue does the following:

1. Checks to see if there is a job, pops it from the queue if so (uses reentrant locks for thread safety).

2. If a job is taken, it runs the job's rendering function.

3. Otherwise, the worker thread will give up execution for a short period of time (sleep) and repeat this process.

When a layer pushes a tile onto the queue, it first checks if there are any duplicate jobs and removes old ones accordingly. Duplicate detection is done by matching IDs (strings) as specified by layers for individual tiles. In some cases (such as in the World Wind layer), it is desirable to remove a large set of jobs from the queue. For instance when we zoom in, in the World Wind layer we no longer need to load tiles from any other zoom level. This is handled by the "invalidate" function which removes all tiles from the queue when necessary. The World Wind tiles' IDs start with "data/worldwindcache", and giving that prefix to the invalidate function would remove all World Wind tiles from the queue.

Chapter 4

Results and Evaluation

In this chapter, we present the output of MarineVis and then we compare the uncertainty visualization between MarineVis and existing visualization approaches.

4.1 Visualization using MarineVis

In our MarineVis system (figure 4.1), on the left side is the graphics viewport which covers most of the display, showing primarily the NASA World Wind terrain map. The bar on the top of the viewport of MarineVis displays the current longitude and latitude of the mouse cursor. The viewport is navigated much in the same way as a web map such as Google Maps. Dragging the mouse using the left mouse button pans the map, the mouse wheel is used to zoom in/out focused on the mouse pointer. On the right of the MarineVis system, layer options are displayed. In general, each set of layer options consists of:

1. A button to hide or show the layer (a check box with the label 'On' or 'Off', to the right of the layer label).

2. A button to minimize the layer options (a button labeled '-' or '+', directly to the right of the check box).

3. A slider to adjust the opacity of the layer in a frame titled 'Layer Opacity'.

We do not retain any options for the NASA World Wind layer because we argue there is no need to hide it or make it translucent. We consider it as base layer which is
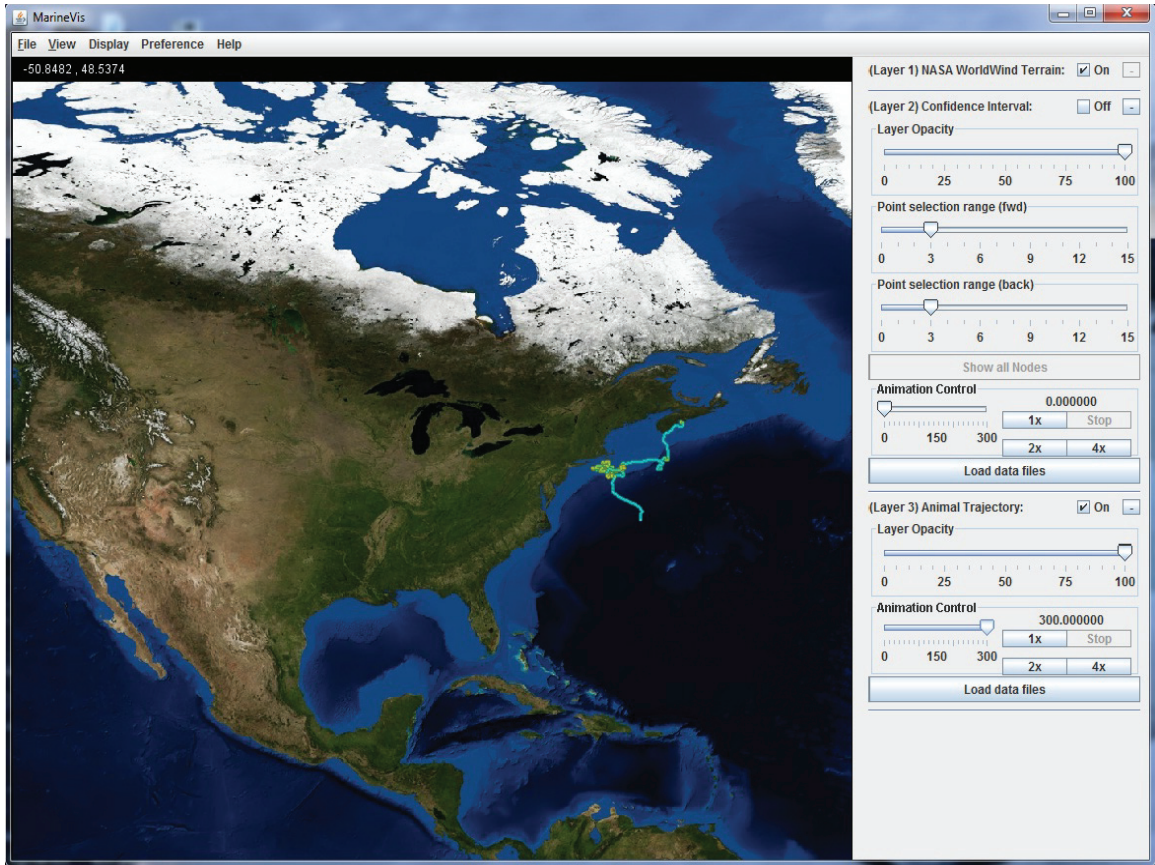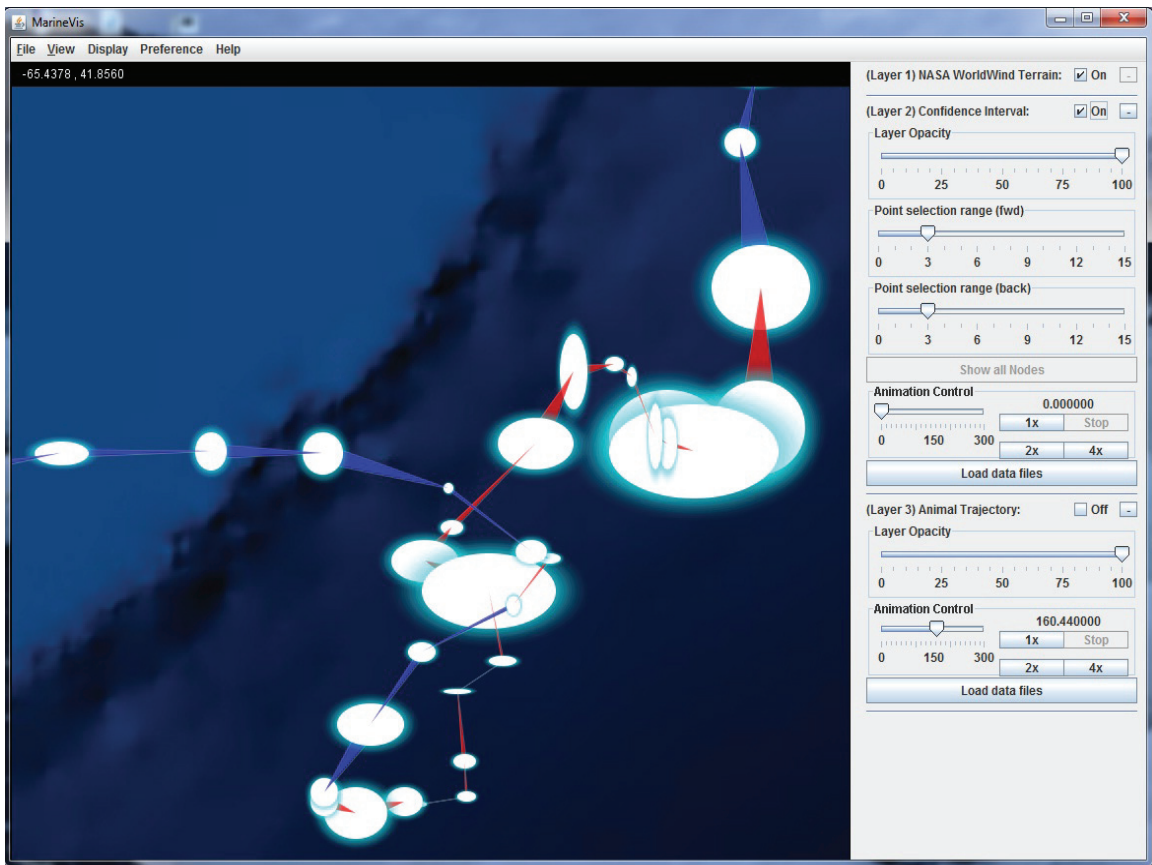
Figure 4.1: MarineVis User Interface

Figure 4.2: Confidence Interval (CI) layer: In this figure, radii of each ellipse represent the credible interval which is generated using SSM code provided by Jonsen et al. [23] and WinBUGs (table 3.1).

always useful to have visible to understand the animals' location. At the top of our system, there are standard menu options such as 'File', 'View', 'Display', 'Preference', and 'Help'.

In the Confidence Interval layer (figure 4.2) of MarineVis, each ellipse is surrounded by an aura, the radii of which represent the standard deviation of longitude and latitude and the color comes from applying a gradient from the 2.5 percentile credible intervals' behavioral mode (table 3.1). A yellow color indicates foraging and a

turquoise color represents migrating behavior of a turtle. Each ellipse is connected to the projection by a triangle. This layer has extra options on the right side from top to bottom: 'Point selection range (fwd)', 'Point selection range (back)', 'Show all Nodes', 'Animation Control', and 'Load data files'. And as the 'Load data files' button name suggests, clicking this button will bring up a prompt to select different files for visualization. When a user selects any particular state/node in the 'Confidence Interval Layer', the color of the node changes and becomes yellow (figure 4.3). Depending on the slider, the number of front and back nodes becomes visible and all other nodes become invisible. The original view again can be brought back by clicking the 'Show all Nodes' button. In figure 4.3, we observe six nodes before and six nodes after the selected node. The slider labeled 'Point selection range (fwd)' allows adjustments of the number of nodes shown after the focused node, forward. The animation control contains four buttons: 'Play', this will cause the focused node to progress to the next one at fixed time intervals, '2x' will play two times faster, '4x' will four times faster and finally 'Stop' will stop the animation. When the turtle tracks cross each other, it is difficult to follow the tracks in IDV while in MarineVis it is more tractable (figure 4.4 and figure 4.6).

In MarineVis, the animal trajectory is colored with the mean of behavioral mode of SSM output (table 3.1). The centre point of each position on the track is defined either from the output of SSM or a linearly interpolated estimation between two estimated locations. The color is defined by the behavioral mode where more turquoise indicates migration and more yellow represents foraging behavior of a turtle. In figure 4.4, and figure 4.5, we can more closely compare the same tracks using MarineVis and IDV. In figure 4.7 and figure 4.8, we see another two tracks where estimated animal trajectories are visualized.

In MarineVis, in the Animal Trajectory layer, the turtle track can be animated, similar to the Confidence Interval layer, with the 'Animation Control' in its layer properties. In our system, the default animation position is 0. The observation index can be selected by dragging the slider. The buttons 'Play', '2x', '4x', have the effect of starting the animation from the current position at normal, two times, and four times speed respectively. In figure 4.9, we see the animation is paused, at the half of the course of a track, in the upper right portion of the viewport. In figure 4.10, we see the node selection capability and in figure 4.11, we see the transparency option in MarineVis.
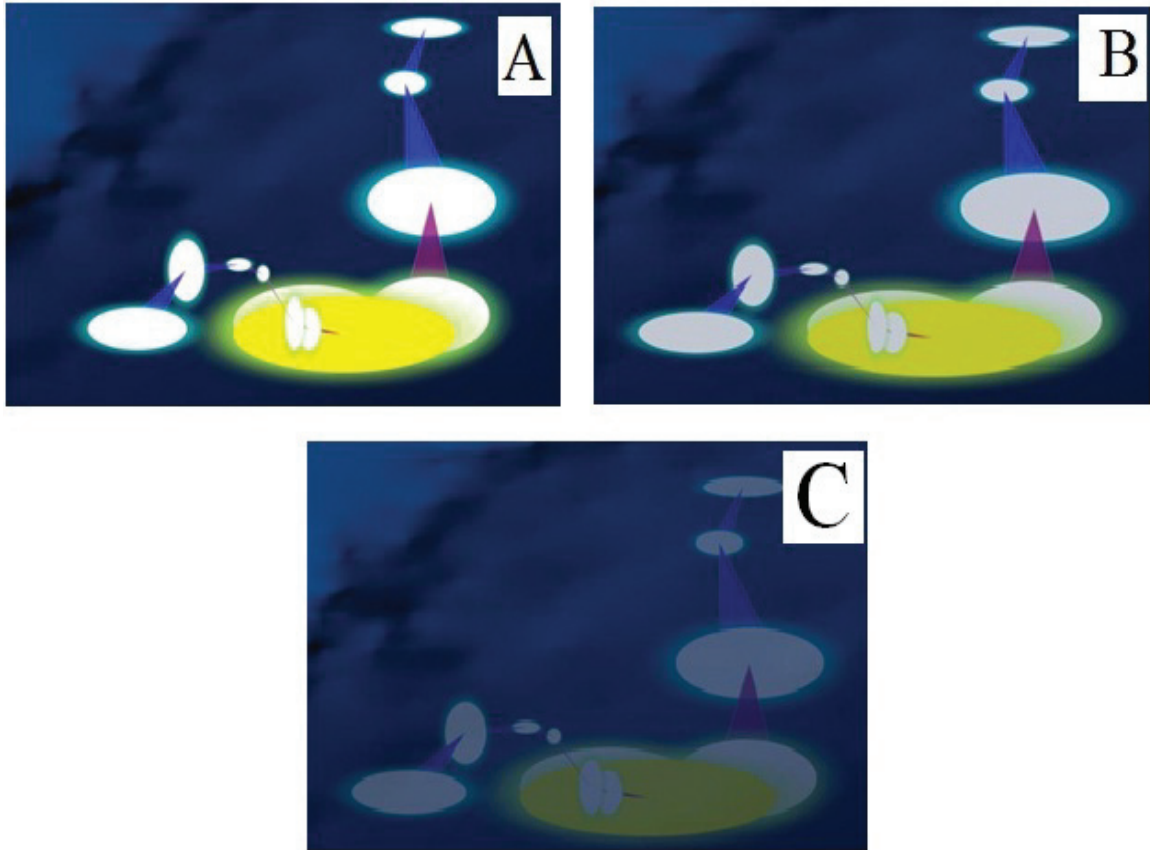
Figure 4.3: Layer opacity: (a) Layer opacity at 100 percent, (b) Layer opacity at 75 percent, and (c) Layer opacity at 25 percent. In each case, six nodes are displayed before and after the focus node.

4.2 Comparing MarineVis with Existing Approaches

Existing software such as IDV, MAMVIS, GeoZui3D or GeoZui4D does not visualize any form of positional or behavioral uncertainty. At first, we had developed components for IDV to visualize the confidence intervals (figure 4.12, and figure 4.13). But several problems emerged. One of them is when tracks cross each other. With our MarineVis system, even when the tracks overlap or cross each other, users can easily understand and follow the tracks which helps them to understand the positional and
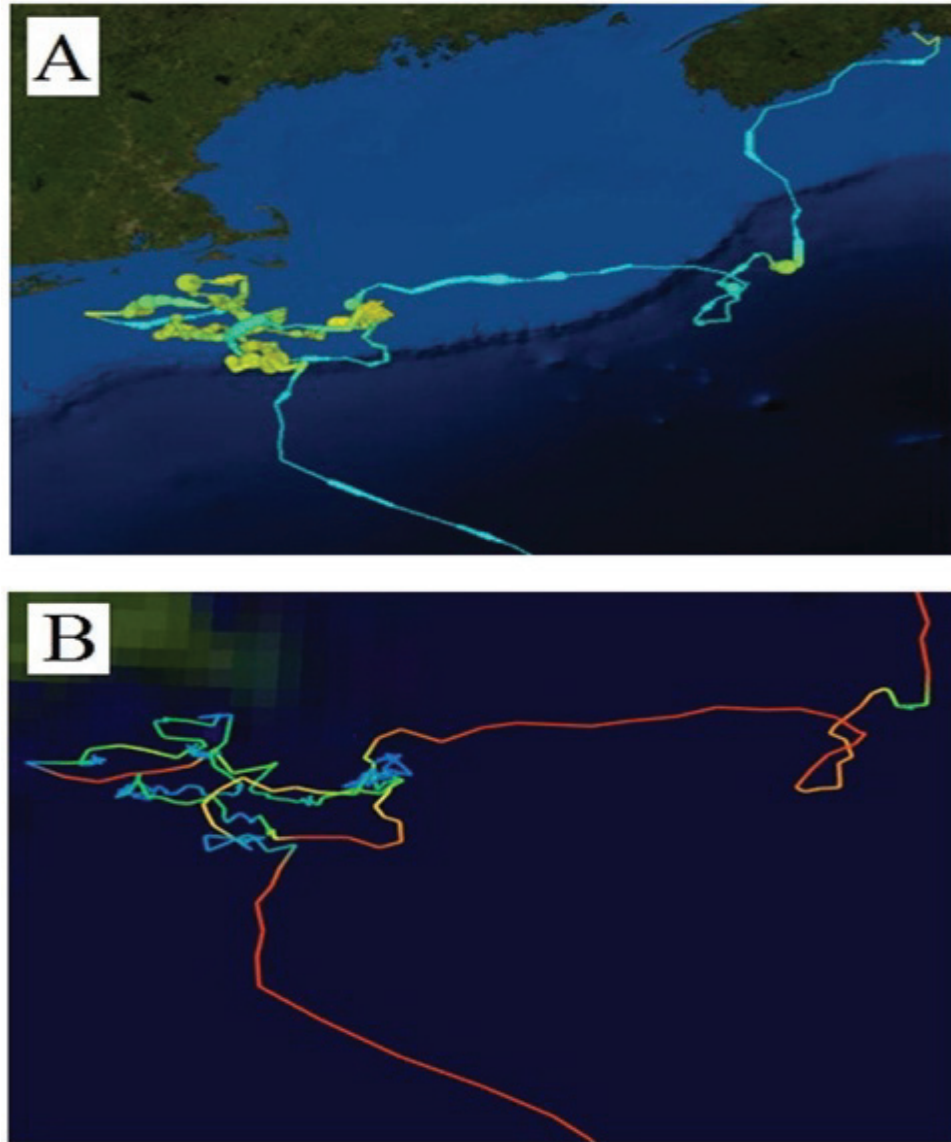
Figure 4.4: Animal trajectory displayed using IDV and MarineVis: This image shows the broad view of the Animal trajectory layer in MarineVis (A) as well as IDV, (B) using the same data file.

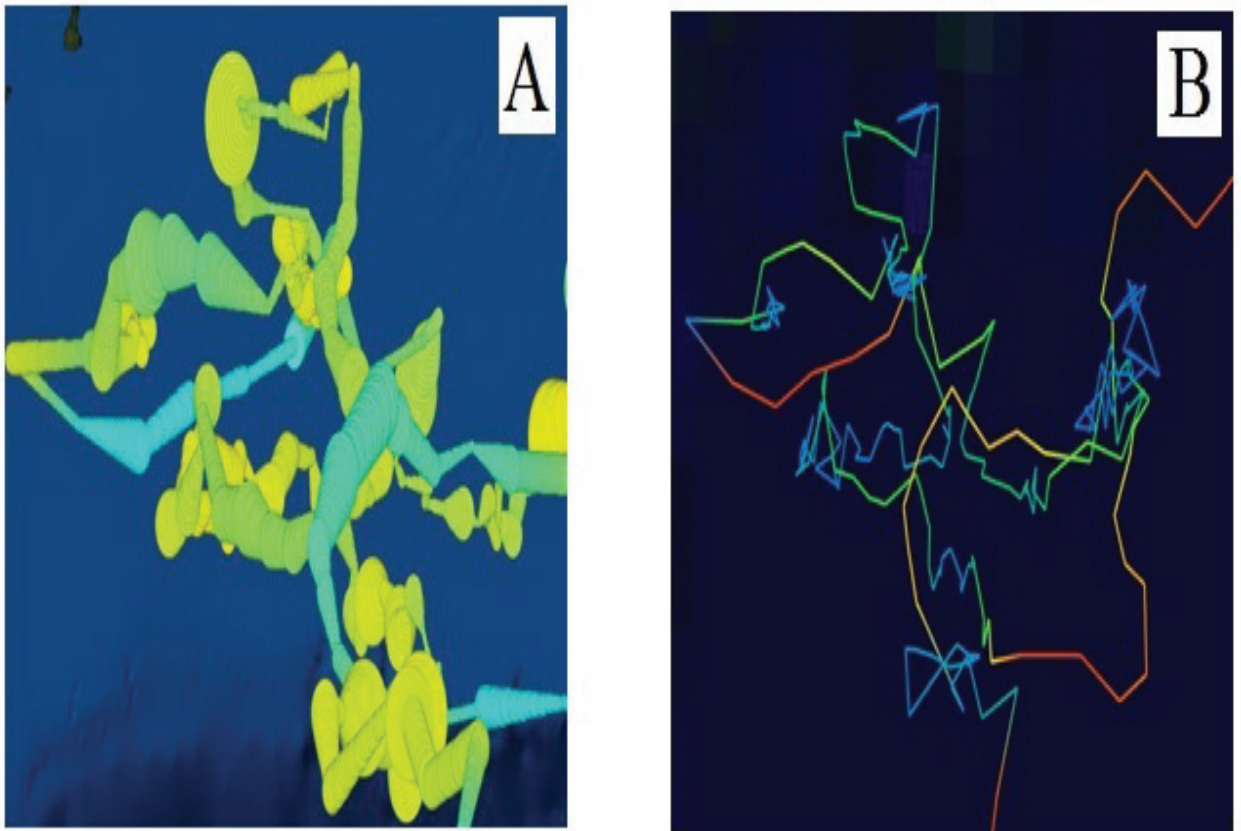Figure 4.5: Comparison of animal trajectory between IDV and MarineVis: This image shows a closer look of the animal trajectory using MarineVis (A) and IDV(B).

Figure 4.6: Closer view of an animal trajectory using MarineVis: This figure shows that even when the intersecting tracks have the same color, users still can easily follow that.

Figure 4.7: Second turtle track: This figure visualizes the estimated track of the second turtle.

Figure 4.8: Third turtle track: This figure visualizes the estimated track of the third turtle.
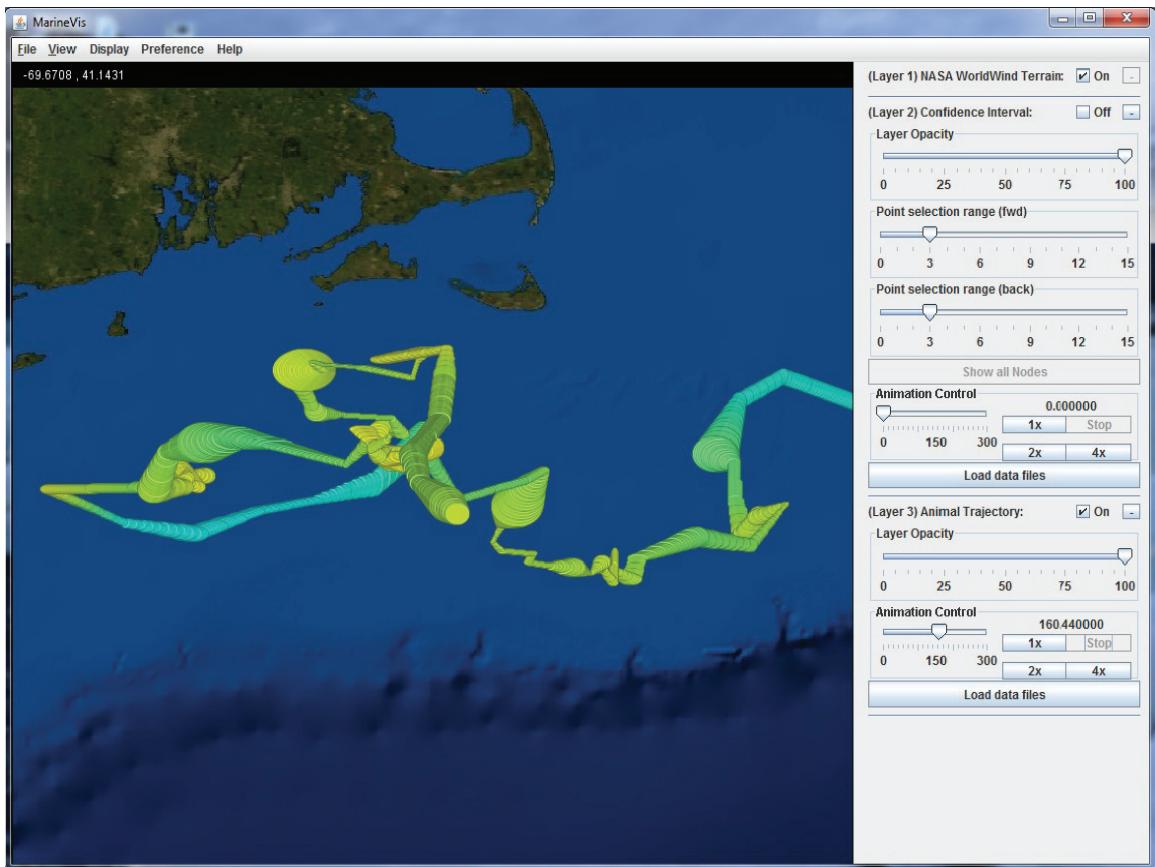
Figure 4.9: MarineVis animation: This figure illustrates the Animal Trajectory layer's animation abilities.
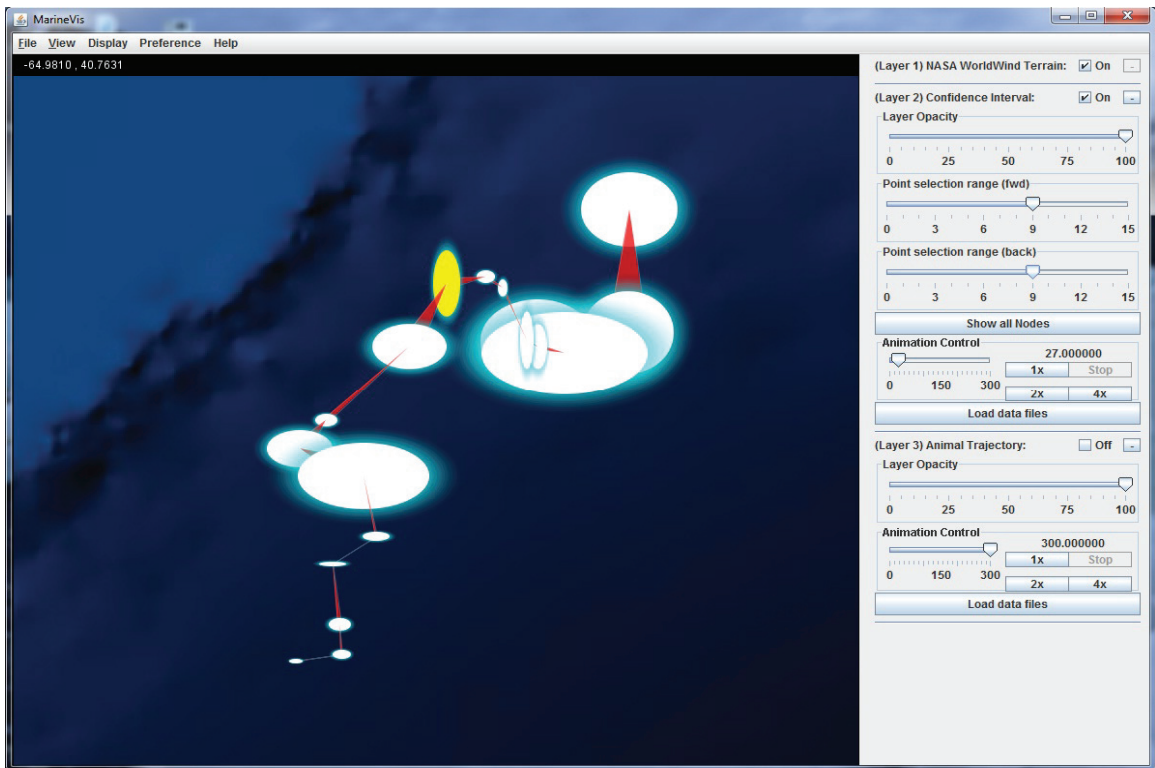
Figure 4.10: Node selection capability of MarineVis: In this figure the point selection range forward is set to 9 (as seen on the settings bar for this layer) and the point selection range backward is set to 9 as well.

Figure 4.11: Transparency in MarineVis: This figure shows how transparency settings can be used to merge layers and produce different useful views. This image shows the Animal Trajectory layer on top, with full opacity, and underneath it, the Confidence Interval layer, with around 25 percent opacity.

behavioral uncertainty of the turtle at the same time (figure 4.6, and figure 4.9).

In current approaches which visualize uncertainty such as in figure 2.5, there is no option for users to interact and interrupt. The output are static images. Users have fewer options to choose or s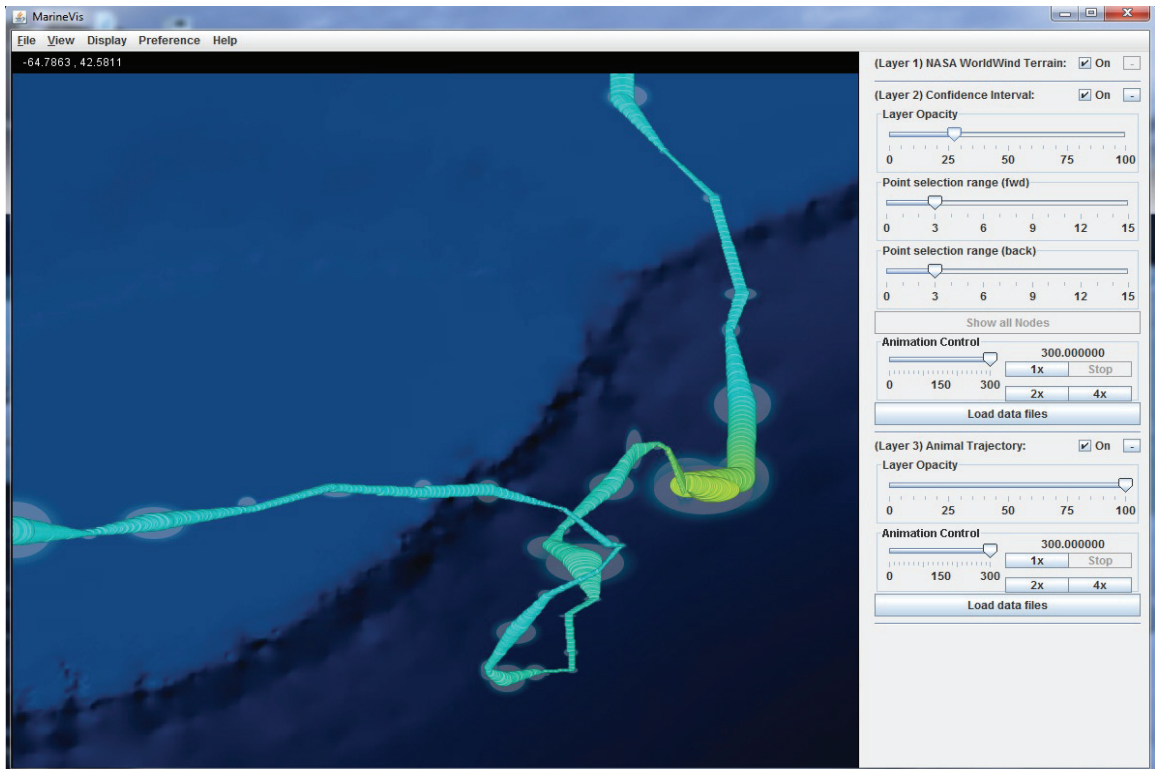elect any particular state of the animals or animate the trajectories of marine animals. There is no transparency option in those systems. Our system has a transparency option which allows users to view individual layers or a specified amount of data. MarineVis does not occlude the underlying terrain as much because the glyphs are localized at the sample points rather than being spread out around the entire path in the Confidence Interval layer. Our system encodes both direction and position than just position. The example visualization shown in figure 2.5 would not work well for self-crossing paths which are common with marine animal data. The ellipses allow a more direct comparison between two data points rather than a continuously changing path width. Users can interactively limit the view of data points to a subset of available data points on a path, in clustered regions, to reduce congestion and can animate the progression of the animal along its trajectory which is absent in existing approaches.
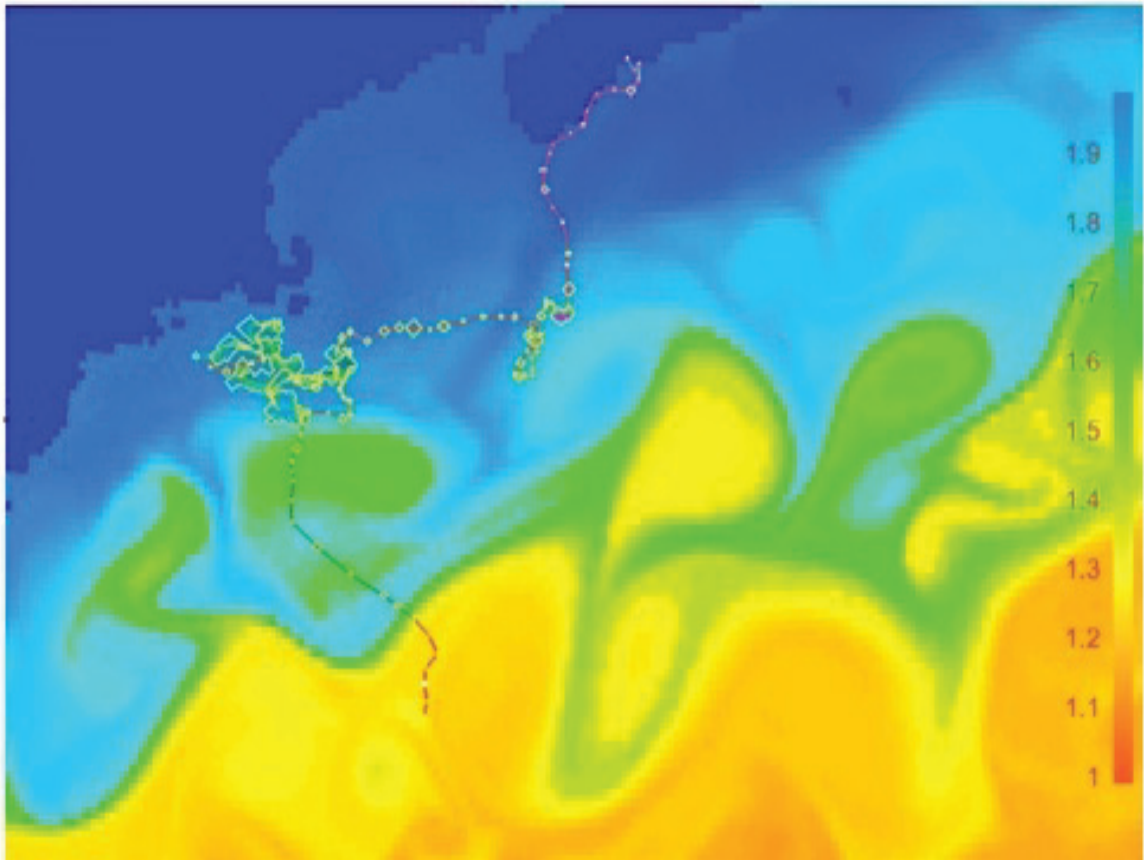
Figure 4.12: Turtle track with sea water elevation: This figure shows a turtle track after applying SSM with sea water elevation using a modified version of IDV.
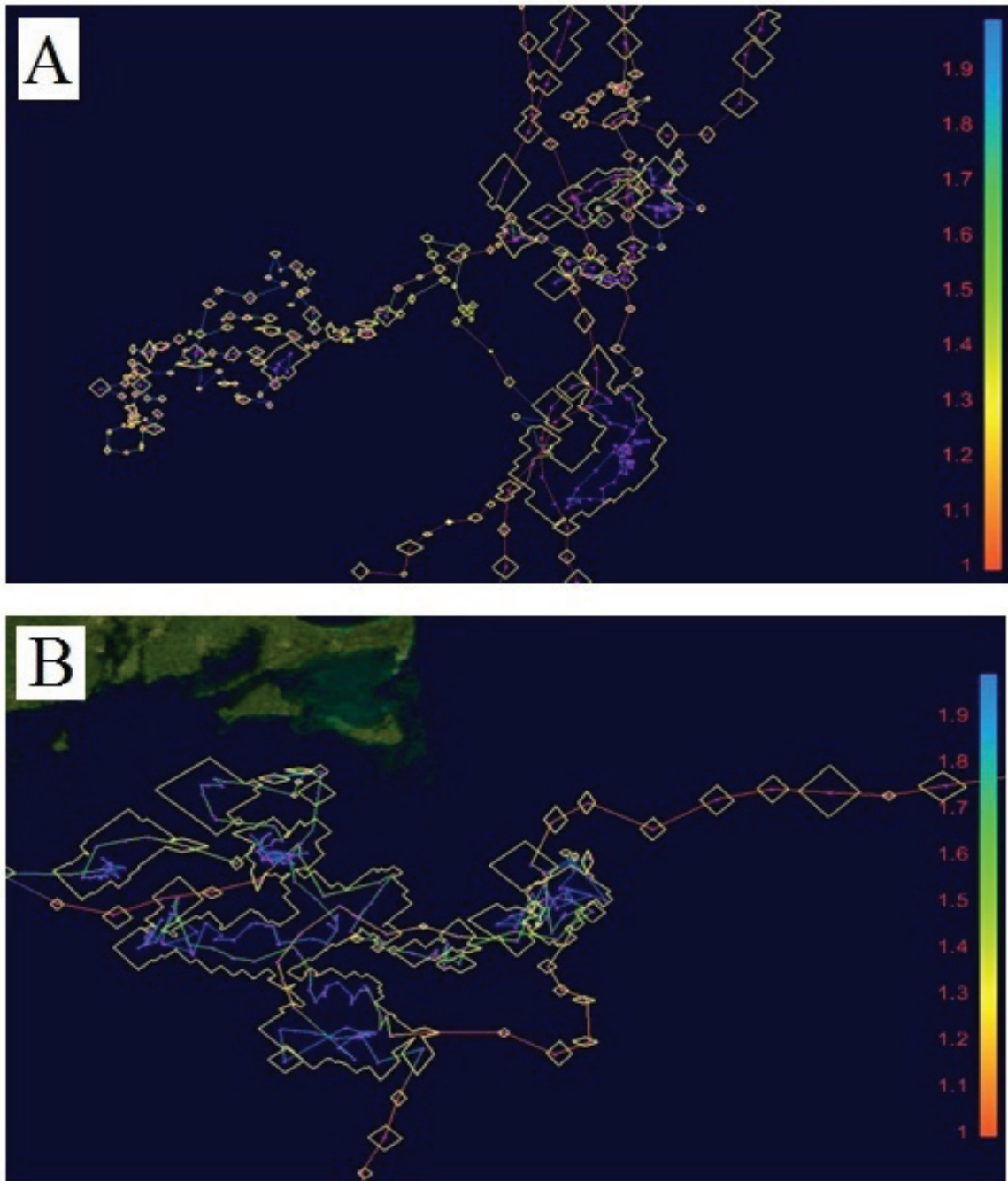
Figure 4.13: Confidence intervals of a turtle track using IDV: These images show two turtle tracks that have been displayed using a modified version of IDV. The track is colored according to the behavioral mode where red indicates migrating and blue indicates foraging. Here we see polygons which are formed by merging areas of uncertainty of the confidence intervals, using an algorithm [51] to perform an 'or' operation on a set of intersecting polygons in both of the figures of A and B.

Chapter 5

Conclusion

Marine animal movement is a fundamental yet poorly understood process. One of the reasons is because our understanding of movement is affected by the measurement error during the observation and process noise. Differentiating real movement behavior from observation error in data remains difficult and challenging. Methods that acknowledge uncertainty in movement pathways when estimating constantly changing animal movement have been lacking until this time. However with the arrival of state-space models, this problem is partially solved as SSMs acknowledge this problem by allowing unobservable true states to be estimated from data observed with errors which arise from imprecise observations. State-space models use Markov Chain Monte Carlo methods which generate samples from a distribution by constructing a Markov Chain where the current state only depends on the immediately preceding state. The task of fitting SSMs to data is challenging and requires large computational effort and expertise in statistics. With the arrival of the WinBUGs software, this formidable task becomes relatively easy. Though using the WinBUGs software researchers try to visualize the tracks and behaviors, new problems appear. One of the problems is that when marine animals come back to certain places or animals' tracks cross each other several times, the tracks become cluttered and users are not able to understand the direction. Another problem of visualizing the confidence intervals generated using SSMs is that images generated using other systems are static in nature and therefore lack interactivity. Information becomes cluttered when too much data appear. Users are not able to differentiate tracks, confidence intervals

or the information they would like to visualize. Acknowledging these, we have designed and implemented an interactive visualization system, MarineVis, where these problems are overcome. Using our system the confidence intervals generated using the SSMs, can be visualized more clearly and the direction of the turtle tracks can be understood easily. Our system does not occlude the underlying terrain as much because the glyphs are localized at the sample points rather than being spread out around the entire path. Our system encodes both direction and position rather than just position. Users can interactively limit the view of data points as a subset of available data points on a path, in clustered regions, to reduce congestion, and can animate the progression of the animal along its trajectory which is absent in existing approaches. All these results are visualized over NASA World Wind maps that facilitates the understanding of the tracks.

Although our system is able to visualize marine animal behavior, there is still room for improvement. Two main features are described below which could be incorporated into our system. Firstly, incorporating NetCDF files along with turtle tracks could help scientists understand how environmental factors such as sea water velocity, sea water temperature, and sea water pressure influence turtles behavior. Secondly, incorporating a modeling perspective into our visualization system would enable users to study the influence of different parameters on the estimated turtle tracks.

As mentioned often in the literature that is focused on the analysis of animal movement data [22, 23, 24], incorporating external data linked to the animal tracks by position and time can lead to better analysis of the animals' psychology and biology. The MarineVis application already overlays geographical data along with the path visualizations. Environmental factors such as sea water temperature, sea water

pressure and sea water velocity will aid in understanding and verifying the movement data. Another useful feature would be to allow the addition of NetCDF Climate Forecast (CF) data files. The MarineVis application would be adapted to allow for arbitrary inserting, deleting and/or reordering of layers defined by CF-NetCDF data files. Users would be able to select a data file, a gradient (which could include transparency) to color the data on the map. Users would be able to change all of these attributes or delete a layer after the initial configuration. In order to better aid the ecologist and ethologists, there will be a method to graph the animal track against one or more of these CF layers over time. This would help researchers to compare the animal's time spent in areas of different climate variances and their flight patterns. This would also augment users' understanding of the marine animals' trajectories better.

Also by incorporating a modeling perspective into our visualization system, researchers would be able to easily change different parameters such as process noise, measurement errors for the turtle tracks, which would enable them to clearly understand marine animals' trajectories better.

In future, we plan to release our visualization system to the marine biology community. We strongly believe that our approach of visualization is worthy of further studying. We also hope that our research will influence and inspire others who will be working in the difficult and challenging but amazingly interesting and exciting research field of marine visualization.

Bibliography

[1] K.H. Andersen, A. Nielsen, U.H. Thygesen, H. Hinrichsen, and S. Neuenfeld. Using the particle filter to geolocate atlantic cod (gadus morhua) in the baltic sea, with special emphasis on determining uncertainty. *Canadian Journal of Fisheries and Aquatic Sciences*, 64(4):618–627, 2007.

[2] R. Anderson-Sprecher and J. Ledolter. State-space analysis of wildlife telemetry data. *Journal of the American Statistical Association*, 86(415):596–602, 1991.

[3] R. Arsenault, C. Ware, M. Plumlee, S. Martin, L.L. Whitcomb, D. Wiley, T. Gross, and A. Bilgili. A system for visualizing time varying oceanographic 3D data. In *OCEANS'04. MTTS/IEEE TECHNO-OCEAN'04*, volume 2, pages 743–747. IEEE, 2005.

[4] M.K. Beard, B.P. Buttenfield, and S.B. Clapham. NCGIA research initiative 7: Visualization of spatial data quality. Technical Paper 91-26. *National Center for Geographic Information and Analysis*, 1991.

[5] C.M. Bergman, J.A. Schaefer, and SN Luttich. Caribou movement as a correlated random walk. *Oecologia*, 123(3):364–374, 2000.

[6] B.A. Block, H. Dewar, S.B. Blackwell, T.D. Williams, E.D. Prince, C.J. Farwell, A. Boustany, S.L.H. Teo, A. Seitz, A. Walli, and D. Fudge. Migratory movements, depth preferences, and thermal biology of Atlantic bluefin tuna. *Science*, 293(5533):1310, 2001.

[7] B.J.L. Boeuf, D.P. Costa, A.C. Huntley, and S.D. Feldkamp. Continuous, deep diving in female northern elephant seals, Mirounga angustirostris. *Canadian Journal of Zoology*, 66(2):446–458, 1988.

[8] M.S. Coyne and B.J. Godley. Satellite Tracking and Analysis Tool(STAT): an integrated system for archiving, analyzing and mapping animal tracking data. *Marine Ecology Progress Series*, 301:1–7, 2005.

[9] D. Draper. Assessment and propagation of model uncertainty. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):45–97, 1995.

[10] D.S. Ebert, R.M. Rohrer, C.D. Shaw, P. Panda, J.M. Kukla, and D.A. Roberts. Procedural shape generation for multi-dimensional data visualization. *Computers & Graphics*, 24(3):375–384, 2000.

[11] D.S. Ebert, C.D. Shaw, A. Zwa, and C. Starr. Two-handed interactive stereoscopic visualization. 1996.

[12] M.A. Fedak, P. Lovell, and B.J. McConnell. MAMVIS: A marine mammal behaviour visualization system. *The Journal of Visualization and Computer Animation*, 7(3):141–147, 1996.

[13] N.D. Gershon. Visualization of fuzzy data using generalized animation. In *Proceedings of the 3rd conference on Visualization'92*, pages 268–273. IEEE Computer Society Press, 1992.

[14] W.R. Gilks, S. Richardson, and D.J. Spiegelhalter. *Markov chain Monte Carlo in practice*. Chapman & Hall/CRC, 1996.

[15] M. Goodchild, B. Buttenfield, and J. Wood. Introduction to visualizing data validity. *Visualization in Geographical Information Systems*, pages 141–149, 1994.

[16] P. Goovaerts. *Geostatistics for natural resources evaluation*. Oxford University Press, USA, 1997.

[17] P.J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711, 1995.

[18] T.F. Gross, K.T. Bosley, and K.W. Hess. *The Chesapeake Bay Operational Forecast System (CBOFS): technical documentation*. US Dept. of Commerce, National Oceanic and Atmospheric Administration, National Ocean Service, Office of Coast Survey, Coast Center for Operational Oceanographic Products and Services, 2000.

[19] I. Hanski, C. Erãlahti, M. Kankare, O. Ovaskainen, and H. Sirén. Variation in migration propensity among individuals maintained by landscape structure. *Ecology Letters*, 7(10):958–966, 2004.

[20] B. Hibbard. Visad. Website accessed: March, 2011, `http://www.ssec.wisc.edu/~billh/visad.html`.

[21] W. Hibbard, C. Rueden, S. Emmerson, T. Rink, D. Glowacki, T. Whittaker, D. Murray, D. Fulker, and J. Anderson. Java distributed objects for numerical visualization in VisAD. *Communications of the ACM*, 45(4):160–170, 2002.

[22] I.D. Jonsen, J.M. Flemming, and R.A. Myers. Robust state-space modeling of animal movement data. *Ecology*, 86(11):2874–2880, 2005.

[23] I.D. Jonsen, R.A. Myers, and J.M. Flemming. Meta-analysis of animal movement using state-space models. *Ecology*, 84(11):3055–3063, 2003.

[24] I.D. Jonsen, R.A. Myers, and M.C. James. Robust hierarchical state–space models reveal diel variation in travel rates of migrating leatherback turtles. *Journal of Animal Ecology*, 75(5):1046–1057, 2006.

[25] P.M. Kareiva and N. Shigesada. Analyzing insect movement as a correlated random walk. *Oecologia*, 56(2):234–238, 1983.

[26] H. Kokko and A. Lopez-Sepulcre. From individual dispersal to species ranges: perspectives for a changing world. *Science*, 313(5788):789, 2006.

[27] S.K. Lodha, A. Pang, R.E. Sheehan, and C.M. Wittenbrink. UFLOW: Visualizing Uncertainty in Fluid Flow. 1996.

[28] A. Lopes and K.W. Brodlie. Accuracy in 3d particle tracing. *Mathematical Visualization: Algorithms, Applications and Numerics*, pages 329–341.

[29] D.J. Lunn, A. Thomas, N. Best, and D. Spiegelhalter. WinBUGS-a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing*, 10(4):325–337, 2000.

[30] A.M. MacEachren. Visualizing uncertain information. *Cartographic Perspective*, 13(3):10–19, 1992.

[31] B.J. McConnell, C. Chambers, and M.A. Fedak. Foraging ecology of southern elephant seals in relation to the bathymetry and productivity of the Southern Ocean. *Antarctic Science*, 4(04):393–398, 1992.

[32] M. Monmonier. Strategies for the interactive exploration of geographic correlation. In *Proceedings of the 4th International Symposium on Spatial Data Handling*, volume 1, pages 512–521, 1990.

[33] M. Monmonier. Time and motion as strategic variables in the analysis and communication of correlation. In *Proceedings: 5th International Symposium on Spatial Data Handling: Aug. 3-7, 1992, Charleston, SC, USA*, page 72. Univ of South Carolina, 1992.

[34] J.M. Morales and S.P. Ellner. Scaling up animal movements in heterogeneous landscapes: the importance of behavior. *Ecology*, 83(8):2240–2247, 2002.

[35] K.B. Newman. State-space modeling of animal movement and mortality with application to salmon. *Biometrics*, 54(4):1290–1314, 1998.

[36] O. Ovaskainen. Habitat-specific movement parameters estimated using mark-recapture data and a diffusion model. *Ecology*, 85(1):242–257, 2004.

[37] A. Pang and A. Freeman. Methods for comparing 3D surface attributes. *SPIE Visual Data Exploration and Analysis*, 1996.

[38] A.T. Pang, C.M. Wittenbrink, and S.K. Lodha. Approaches to uncertainty visualization. *The Visual Computer*, 13(8):370–390, 1997.

[39] T.A. Patterson, L. Thomas, C. Wilcox, O. Ovaskainen, and J. Matthiopoulos. State-space models of individual animal movement. *Trends in Ecology & Evolution*, 23(2):87–94, 2008.

[40] F. Royer, J.M. Fromentin, and P. Gaspar. A state–space model to derive bluefin tuna movement and habitat from archival tags. *Oikos*, 109(3):473–484, 2005.

[41] MODIS satellite-based sensors. Ocean color. Website accessed: March, 2011 , `http://oceancolor.gsfc.nasa.gov/`.

[42] W. Schroeder, K. Martin, B. Lorensen, L.S. Avila, R. Avila, and C.C. Law. *The visualization toolkit*. Prentice Hall PTR, 1998.

[43] C.D. Shaw, J.A. Hall, C. Blahut, D.S. Ebert, and D.A. Roberts. Using shape to visualize multivariate data. In *Proceedings of the 1999 workshop on new paradigms in information visualization and manipulation in conjunction with the eighth ACM internation conference on Information and knowledge management*, pages 17–20. ACM, 1999.

[44] J. Sibert and D. Fournier. Possible models for combining tracking data with conventional tagging data. In *Electronic tagging and tracking in marine fisheries: proceedings of the Symposium on Tagging and Tracking Marine Fish with Electronic Devices, February 7-11, 2000, East-West Center, University of Hawaii*, page 443. Kluwer Academic Pub, 2001.

[45] H. Tanizaki. Estimation of unknown parameters in nonlinear and non-Gaussian state-space models. *Journal of Statistical Planning and Inference*, 96(2):301–323, 2001.

[46] H. Tanizaki. Nonlinear and non-Gaussian state-space modeling with Monte Carlo techniques: A survey and comparative study. *Handbook of Statistics*, 21, 2001.

[47] B.N. Taylor. *Guidelines for Evaluating and Expressing the Uncertainty of NIST Measurement Results (rev)*. DIANE Publishing, 2009.

[48] P. Turchin. *Quantitative analysis of movement: measuring and modeling population redistribution in animals and plants*. Sinauer Associates, 1998.

[49] G. Turk and D. Banks. Image-guided streamline placement. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 453–460. ACM, 1996.

[50] Unidata. Netcdf. Website accessed: March, 2011, `http://www.unidata.ucar.edu/software/netcdf/`.

[51] B.R. Vatti. A generic solution to polygon clipping. *Communications of the ACM*, 35(7):56–63, 1992.

[52] C. Vincent, B.J. Mcconnell, V. Ridoux, and M.A. Fedak. Assessment of Argos location accuracy from satellite tags deployed on captive gray seals. *Marine Mammal Science*, 18(1):156–166, 2002.

[53] C. Ware, R. Arsenault, M. Plumlee, and D. Wiley. Visualizing the underwater behavior of humpback whales. *IEEE Computer Graphics and Applications*, pages 14–18, 2006.

[54] C.M. Wittenbrink. IFS fractal interpolation for 2D and 3D visualization. In *Proceedings of the 6th conference on Visualization'95*, page 77. IEEE Computer Society, 1995.

[55] C.M. Wittenbrink, A.T. Pang, and S.K. Lodha. Glyphs for visualizing uncertainty in vector fields. *Visualization and Computer Graphics, IEEE Transactions on*, 2(3):266–279, 2002.

Appendix A

Process of Loading New Tiles

1. The zoom level is calculated (integer: 0...4).

2. The x, y coordinates of the tiles are calculated (non-negative integers).

$$(x, y) = ([(longitude + 180°)/lzts], [(latitude + 90°)/lzts]). \qquad (A.1)$$

$$(wit, hit) = ([width\_of\_viewport/lzts], [height\_of\_viewport/lzts]). \qquad (A.2)$$

(where wit and hit are the width and height of the viewport's bounding box).

3. For each tile

   (a) The potential path of the tile in the disk cache is generated.

   (b) The URL of the tile is generated, in case the tile does not exist in the cache.

   (c) The tile is queued to be loaded.

The zoom level is calculated from the geographic bounding box of the viewport.

Appendix B

Unit Conversion for Dynamically Rendered Layers

As described before, a tile is a bitmap of which each pixel can be mapped to longitude and latitude. For rendering layers, since we are rendering with geo data, we need functions to convert longitude and latitude into tile pixels. The unit conversion consists of four functions

1. calcX (longitude) : converts a longitude into an x coordinate (tile pixels).

2. calcY (latitude) : converts a latitude into a y coordinate (tile pixels).

3. calcSizeX (longitude_size): converts a horizontally measured geographical size into a size in tile pixels.

4. calcSizeY (latitude_size) : converts a vertically measured geographical size into a size in tile pixels.

The methods calcX and calcY work by first finding the offset from the top left point of the tile in lat/lon to the coordinate component to be converted. This value is then divided by the width or height of the tile, and then multiplied by the width or height of the tile in pixels. The methods calcSizeX and calcSizeY work by first dividing the size by the width or height of the tile and then multiplied by the width or height of the tile in pixels. Similar concepts have been used in the tile rendering function.