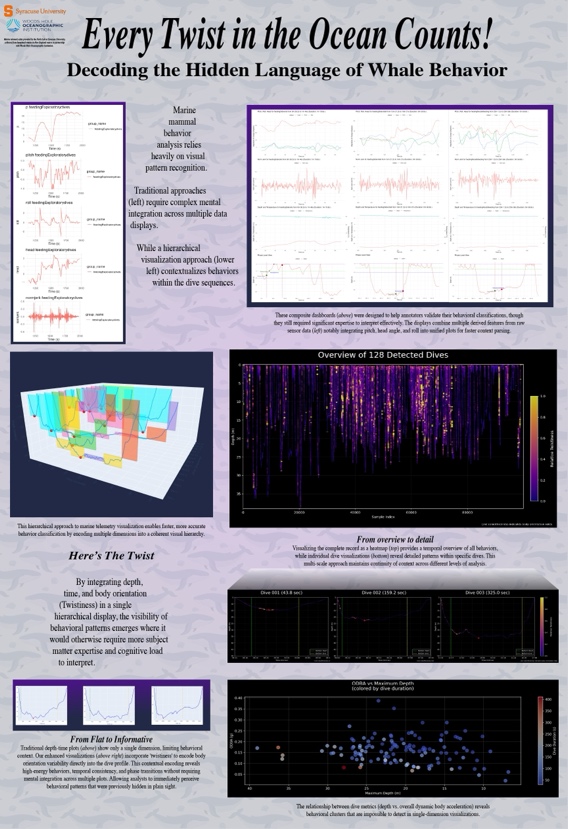
Dynamic Trajectory Visualization: Marine Animal Behavior Analysis

IST 719 Information Visualization

Gil Raitses



# Whale Behavior Visualization: Code Appendix

## Introduction

This code appendix contains R scripts for analyzing and visualizing marine telemetry data, specifically focusing on whale dive behavior. The code demonstrates the evolution of visualization techniques from basic composite dashboards to integrated hierarchical visualizations. All key visualizations use a dark-themed approach to enhance contrast and visual appeal.

Poster: Dynamic Trajectory Visualization

## Visualization Inventory

|  | Visualization Type | Description | Input Data | Output File |
| --- | --- | --- | --- | --- |
| 1 | Composite Dashboard | Early dashboard approach with multiple separate panels | CSV files (mn09\_203a.csv) | output/01\_composite\_dashboard.png |
| 2 | Dive Overview | Heatmap overview of all detected dives | HDF5 file (dive\_analysis.h5) | output/02\_dive\_overview.png |
| 3 | Individual Dive Frames | Series of individual dive visualizations with twistiness encoding | HDF5 file (dive\_analysis.h5) | output/03\_dive\_frames/dive\_\*.png |
| 4 | Hierarchical 3D Visualization | 3D visualization with multiple dives and integrated twistiness | HDF5 file (dive\_analysis.h5) | output/04\_hierarchical\_visualization.html |
| 5 | Energetics Analysis | Comparison of dive depth vs. overall dynamic body acceleration (ODBA) | HDF5 file (dive\_analysis.h5) | output/05\_energetics\_analysis.png |

## 

## Code Structure

The R code is organized into modular scripts for better maintainability:

### Module 1: Setup and Environment (01\_setup.R)

Functions for setting up the environment, creating directories, and utility functions

### Module 2: Data Loading (02\_data\_loading.R)

Functions for loading both CSV and HDF5 file formats

### Module 3: Composite Dashboard (03\_composite\_dashboard.R)

Functions for creating the early multi-panel dashboard approach

### Module 4: Dive Overview (04\_dive\_overview.R)

Functions for creating heatmap overview of all detected dives

### Module 5: Dive Frames (05\_dive\_frames.R)

Functions for creating individual dive frames with twistiness encoding

### Module 6: Hierarchical Visualization (06\_hierarchical\_visualization.R)

Functions for creating the 3D hierarchical visualization with integrated orientation data

### Module 7: Energetics Analysis (07\_energetics\_analysis.R)

Functions for calculating ODBA and creating energetics visualizations

### Main Script: Driver Script (run\_all.R)

The main script that orchestrates the entire visualization pipeline

## Package Requirements

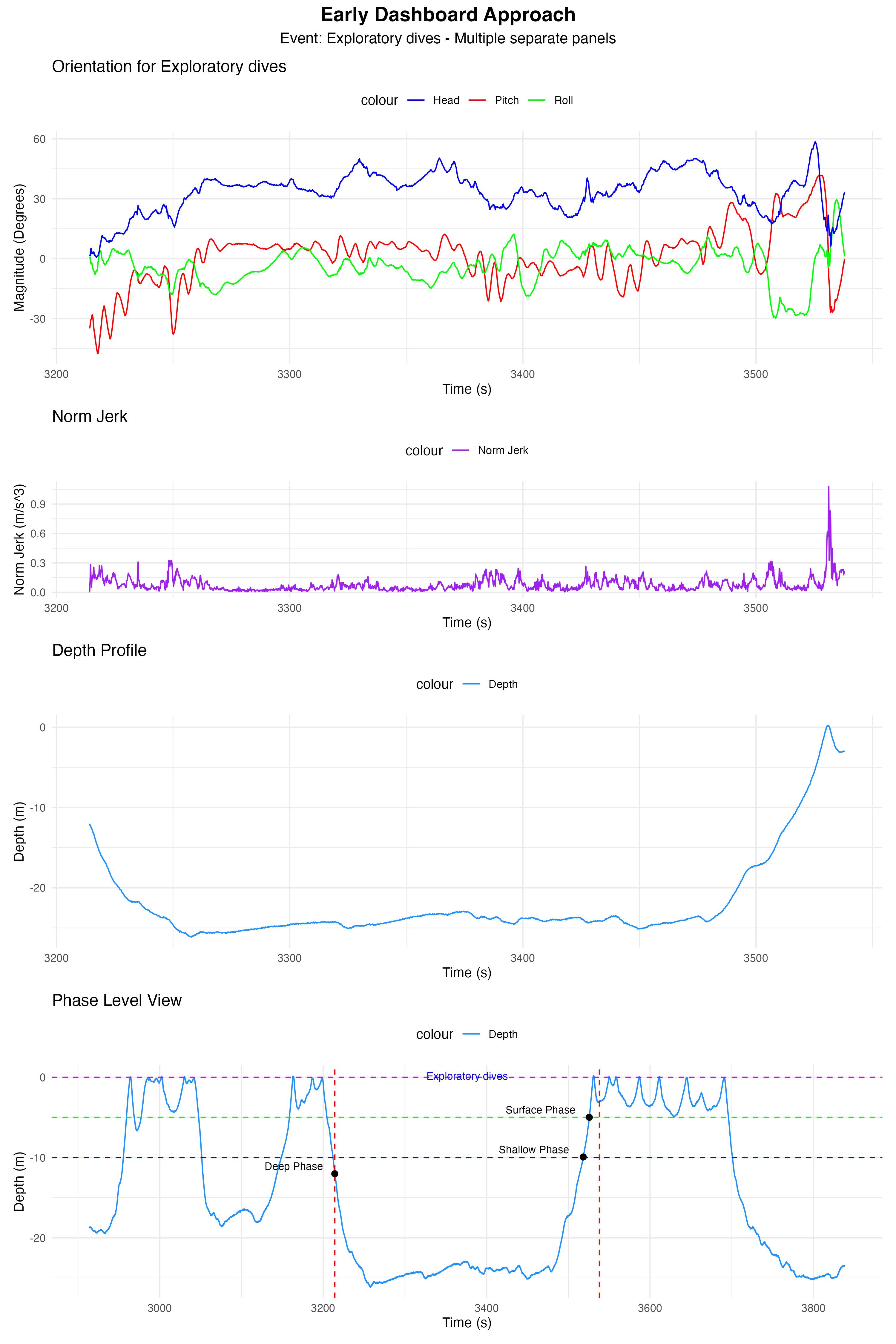
The visualizations require several R packages for different aspects of the pipeline. For general plotting and data manipulation, the code relies on ggplot2, dplyr, viridis, and patchwork. The interactive visualization components use plotly and htmlwidgets to create and save 3D web-based visualizations. For data processing, the pipeline incorporates rhdf5 for reading HDF5 files, signal for processing orientation data, and scales for axis scaling and normalization. All these packages are loaded at the beginning of the pipeline to ensure proper functionality.

## Visualization Descriptions and Outputs

### Visualization 1: Composite Dashboard

**Purpose**: Early approach with multiple separate panels requiring mental integration.

This dashboard provides a comprehensive overview by displaying orientation data (pitch, roll, heading) in the top panel, norm jerk (acceleration) in the middle panel, and depth profile in the bottom panel. While informative, this visualization requires viewers to mentally integrate information across separate panels.

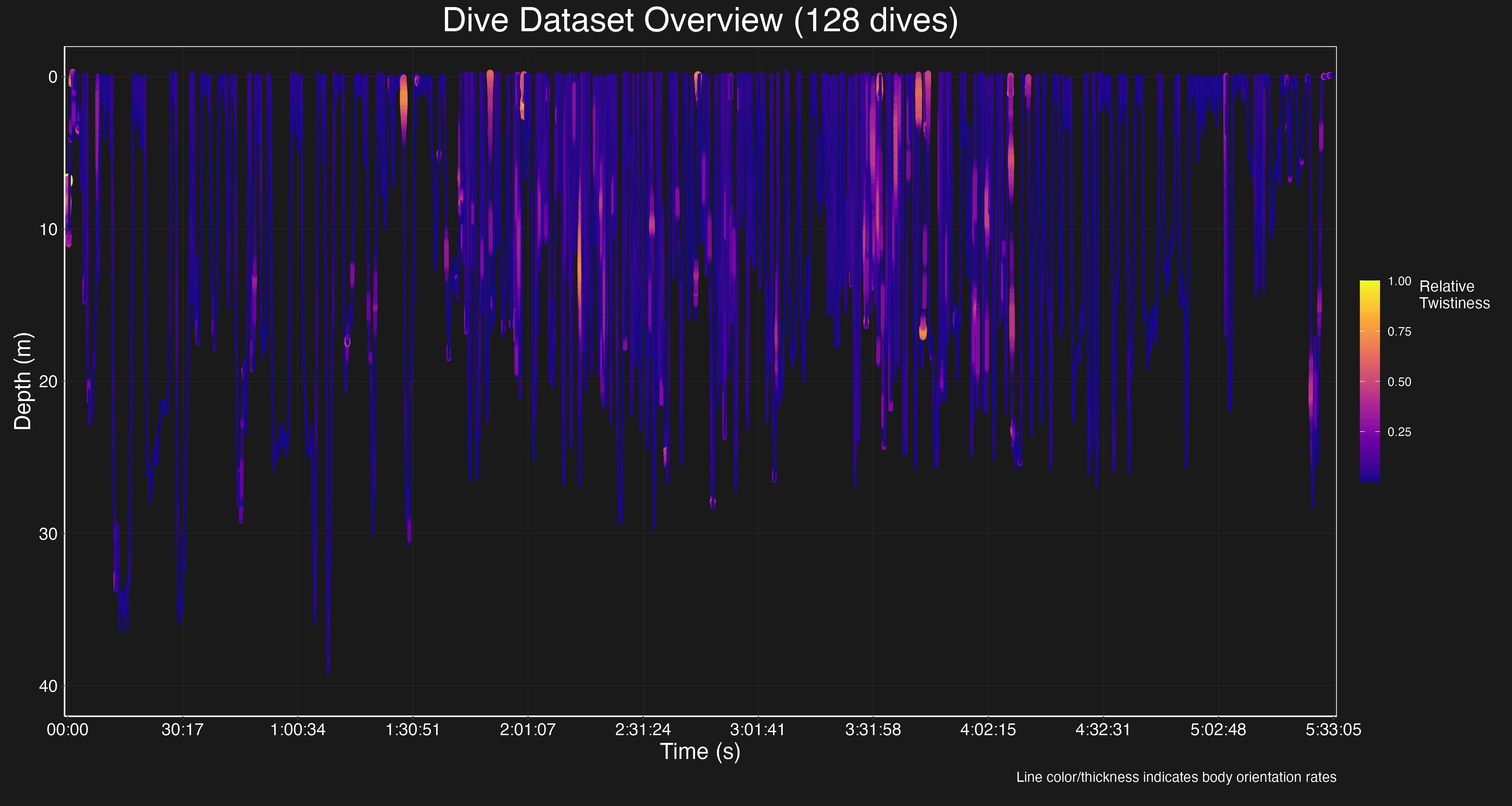


Composite Dashboard

### Visualization 2: Dark-Themed Dive Overview

**Purpose**: Dark-themed visualization providing temporal context across all dives with twistiness encoding.

This visualization creates a depth profile of all detected dives on a dark gray background, with orientation changes (“twistiness”) encoded through both color and line thickness. Dark blue and thin lines indicate minimal orientation changes, while bright yellow and thick lines represent rapid orientation changes. Green and yellow dashed vertical lines mark bottom start and end phases.

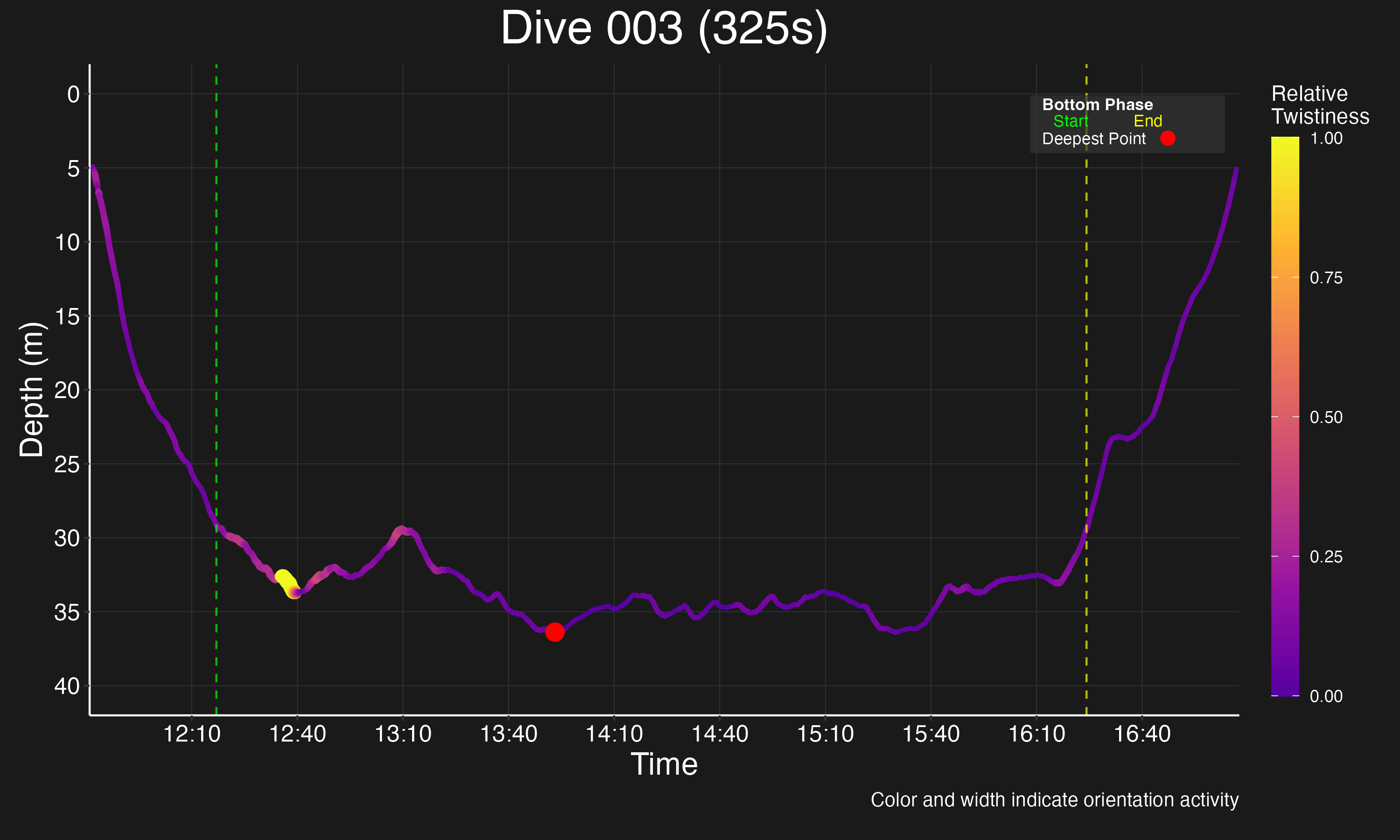


Dive Overview

### Visualization 3: Dark-Themed Individual Dive Frames

**Purpose**: Individual dive visualizations on dark background with orientation changes encoded through color and line thickness.

These visualizations provide detailed information about each detected dive, with dual-channel encoding of orientation changes. The color gradient ranges from blue to yellow (plasma colormap), while line thickness varies from 1.0px to 4.0px. Key reference markers include green dashed lines for bottom phase start, yellow dashed lines for bottom phase end, and red circles for maximum depth points.

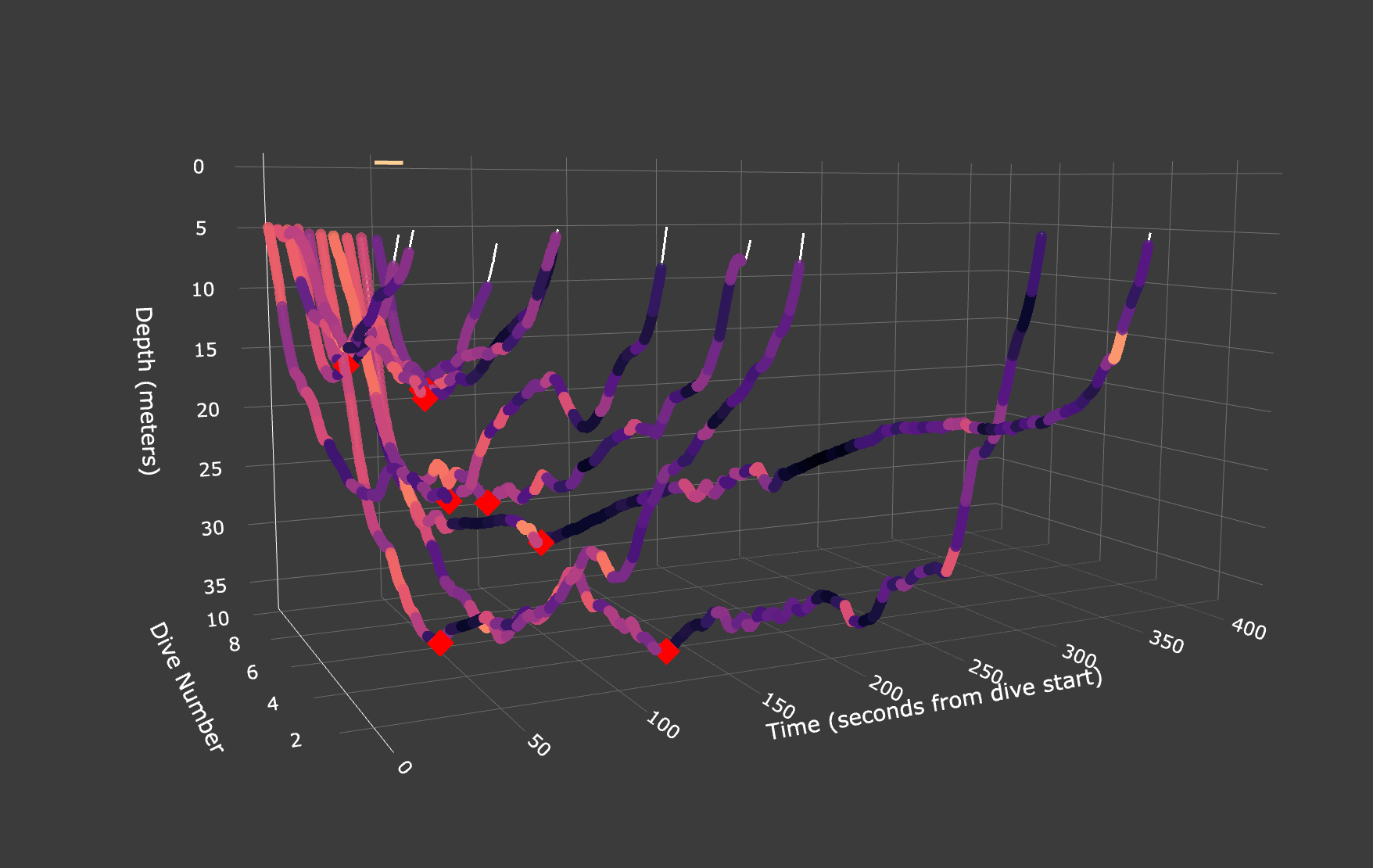


Example Dive Frame

### Visualization 4: Hierarchical 3D Visualization

**Purpose**: Integrated 3D visualization linking depth, time, and orientation.

This interactive visualization creates a 3D plot with time on the x-axis, dive number on the y-axis, and depth on the z-axis. Twistiness is encoded through both color and line width. The output is an interactive HTML file that allows rotation, zooming, and selection, reducing cognitive load by integrating multiple information dimensions.



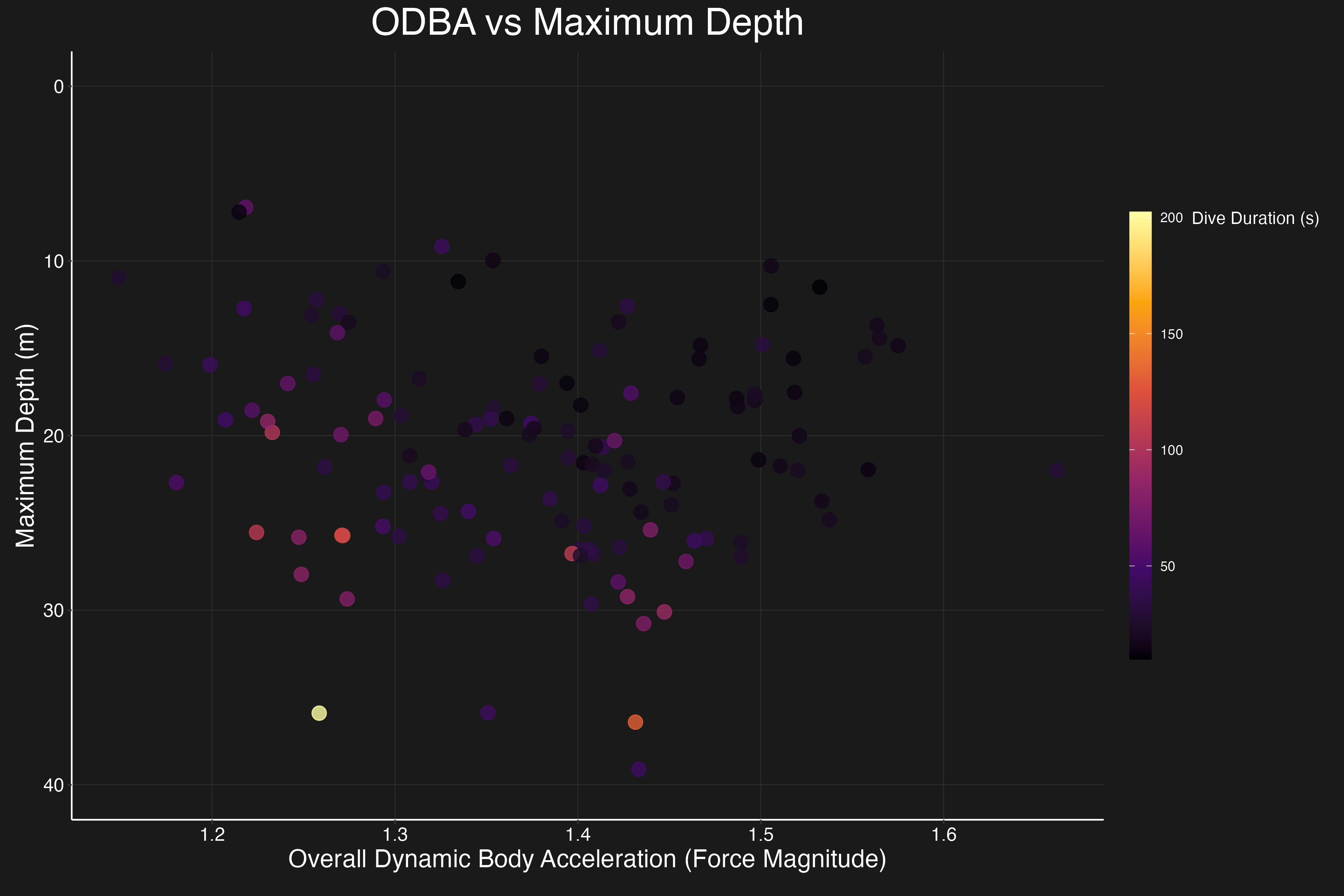
Hierarchical 3D Visualization (Screenshot)

### 

### Visualization 5: Dark-Themed Energetics Analysis

**Purpose**: Analysis of ODBA vs Maximum Depth relationship with dive duration encoding.

This focused visualization examines the relationship between dive depth and energy expenditure (ODBA), with dive duration encoded through color. It implements an inverted x-axis so depth increases intuitively to the right, and uses the inferno colormap to represent dive duration. The dark gray background enhances visual contrast.



Energetics Analysis

## Technical Implementation Details

### Data Processing

#### Dive Detection

The dive detection algorithm identifies dive segments by finding periods where depth exceeds a minimum threshold (default 5m), ensuring dive duration exceeds a minimum time period (default 30 samples), identifying the maximum depth point within each dive, and returning a matrix of start, max\_depth, and end indices for each detected dive.

#### Dive Phase Segmentation

The phase segmentation identifies key transitions in each dive by detecting the bottom phase using a depth threshold (80% of maximum depth), calculating descent and ascent rates, and providing clear phase boundaries for visualization markers.

#### Twistiness Calculation

Twistiness measures the rate of change in body orientation by calculating the absolute derivative of pitch, roll, and heading angles, applying higher weighting to heading changes (factor of 2), using robust normalization based on percentiles (1st to 99th), and providing values in the 0-1 range for visualization encoding.

#### Energetics Analysis

Dynamic Body Acceleration metrics include Overall Dynamic Body Acceleration (ODBA), which measures the sum of absolute acceleration across all three axes and serves as a proxy for energy expenditure, and Vector Dynamic Body Acceleration (VDBA), which measures the vector magnitude of acceleration as an alternative energy expenditure metric.

### Visualization Techniques

#### Dark-Themed Visualization

All key visualizations implement a dark-themed approach with a dark gray background (#1A1A1A) that provides enhanced contrast with visualized data, reduced eye strain during analysis, optimal visibility for both high and low twistiness states, and a professional appearance for presentation contexts.

#### Dual Encoding of Body Orientation

In both dive overview and individual dive frames, twistiness is encoded through two visual channels: color encoding using the plasma colormap (blue/purple for low twistiness, orange/yellow for high twistiness) and line thickness encoding (thin lines for low twistiness areas, thick lines for high twistiness areas).

#### Dive Phase Visualization

Bottom phase markers are implemented as vertical lines, providing clear indicators of descent-to-bottom transition (green dashed line), bottom-to-ascent transition (yellow dashed line), and visual references for phase analysis.

## Usage Instructions

To run the visualization pipeline:

1. Open a terminal window
2. Navigate to the scripts directory
3. Run the driver script with: Rscript run\_all.R
4. Outputs will be saved to the ../output folder (created automatically if it doesn’t exist)

Note: The script uses relative paths to access data in the ../data/ directory.