Title of the Thesis

Subtitle if needed

Author Name

Department, University

Month Year

Abstract

In professional cycling, athletes are often sent to training camps in locations such as the French Alps, Andorra, Sierra Nevada, Colorado Springs, or other high altitude destinations. While a big part of this is altitude acclimatisation, many cyclists report that the biggest effect is actually the suitability of the roads for training. Long, car-devoid mountains, where athletes can just put their head down and focus on their effort means that their training quality is improved, as opposed to having to ride through small villages, stopping at intersections and being constantly vigilant of overtaking cars or traffic furniture. This thesis explores the possibility of being able to track and quantize "Training Suitability" of roads. What is proposed is an integrated framework that can semantically partition continuous trajectory data from a cyclist's GPS device, and meaningfully define each segment in the context of training suitability. Starting from raw GPS traces, we apply geometric filtering and map-matching to correct noise and align positions with a reference network. An adaptive segmentation algorithm then identifies breakpoints using curvature statistics, speed variance, and road metadata (such as road name, or speed limit), yielding segments that have a homogenous quality. Each segment is characterized by spatial, temporal, and contextual features and subsequently classified through a Wavelet Transform function that quantizes variability and changepoints in data streams. The scalability and robustness of the framework will be evaluated through a small-scale live web application.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

1.2 Research Objectives

Chapter 2

Objects of Relevance

Chapter 3

Literature Review

3.1 LSEPI Analysis

3.2 Component A

3.3 Component B

Chapter 4

Requirements Document

4.1 Software Requirements Specification (SRS)

4.1.1 Purpose and Scope

4.1.2 Intended Audience and Reading Suggestions

4.1.3 Overall Description

Product Perspective

Product Functions

User Characteristics

Operating Environment

Design and Implementation Constraints

Assumptions and Dependencies

4.1.4 Specific Requirements

External Interface Requirements

Functional Requirements

Performance Requirements

Logical Database Requirements

Software System Attributes

Reliability

Availability

Security

Maintainability

Portability

Other Requirements

4.2 Standards and Compliance

Chapter 5

Methodology

5.1 Design

5.1.1 Scope and Purpose

Define the objectives and deliverables of the design phase, linking back to the SRS.

5.1.2 Architectural Design

Reference Architecture

Present a high-level system architecture diagram using UML or SysML.

Architectural Views

- **Context View**: System boundaries, external entities, and interfaces.
- **Functional View**: Key modules and their responsibilities.
- **Physical View**: Hardware deployment and network topology.
- **Information View**: Data flow and storage structures.
- **Behavioral View**: Sequence and state diagrams for critical scenarios.

Design Rationale

Justify architectural decisions, discuss alternatives, and document trade-offs.

Design Patterns and Styles

Identify and describe any applied patterns (e.g., MVC, layered) and coding conventions.

5.1.3 Module-Level Design

Module Decomposition

List all system modules, their interfaces, and dependencies. Include a module dependency diagram.

Module Specifications

For each module:

Name: Purpose and description.

Interfaces: Inputs, outputs, and protocols.

Behavior: Algorithmic overview or pseudocode.

Dependencies: Internal and external module links.

5.1.4 Interface Design

User Interface

Wireframes or mockups, navigation flows, and accessibility guidelines (e.g., WCAG 2.1).

Software Interfaces

API contracts, message schemas (e.g., JSON, XML), and error handling.

Hardware Interfaces

Detailed electrical and protocol specifications for device interactions.

5.1.5 Data Design

Data Models

ER diagrams or class diagrams showing data entities and relationships.

Database Schema

Detailed table definitions, keys, indexes, and normalization rules.

Data Dictionary

Definitions and formats for all data elements used in the system.

5.1.6 Security and Safety Design

Risk assessment, threat models, and mitigation strategies.

5.1.7 Standards and Compliance in Design

List all applicable ISO/IEC, IEEE, and domain-specific standards adhered to (e.g., ISO/IEC 27001 for security).

5.2 Implementation

5.3 Testing and Results

Chapter 6

Results and Conclusions

Chapter 7

Further Discussions and Research Gaps

Appendix A

Appendix

A.1 Further Reading

A.2 Source Code

A.3 File Structure

A.4 Additional Documentation