

Architectural design of an internet of things-based framework for road bike speed optimization

Tigor Hamonangan Nasution¹, Opim Salim Sitompul², Fahmi Fahmi¹, Muhammad Anggia Muchtar²

¹Department of Electrical Engineering, Faculty of Engineering, Universitas Sumatera Utara, Medan, Indonesia

²Department of Information Technology, Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Medan, Indonesia

Article Info

Article history:

Received Oct 27, 2025

Revised Mar 10, 2026

Accepted Apr 22, 2026

Keywords:

Edge computing

Internet of things

Machine learning

Optimal speed

Pacing zone

Predictive framework

Road bike

ABSTRACT

This research aims to develop an internet of things (IoT) system framework to predict cyclists' optimal speed in road cycling using multisensor data and machine learning. The primary issue raised is the lack of an intelligent system capable of integrating physiological, performance, and environmental data in real-time speeds for cyclists. The designed framework consists of four functional layers: data acquisition layer; data processing and feature layer; predictive modeling layer; and recommendations and output layer. Modeling is carried out using gradient boosting regression (GBR), performed end-to-end with validation on real cyclist activity data. The test results demonstrate that the system can provide precise optimal speed estimates and offer pacing zone recommendations that positively impact athlete performance strategies. This research contributes novelty in the form of an adaptive multivariate prediction approach and a modular IoT architecture design that can be implemented on cloud and edge platforms.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Tigor Hamonangan Nasution

Department of Electrical Engineering, Faculty of Engineering, Universitas Sumatera Utara

Medan, Indonesia

Email: tigor.nasution@usu.ac.id

1. INTRODUCTION

A road bike is designed for high-speed riding and is widely used by both professional athletes and hobbyists [1]–[3]. In competitive cycling, performance depends not only on physical capacity but also on pacing strategy to manage energy and sustain speed [4]–[6]. However, pacing decisions are still often based on intuition and subjective coaching experience [7], [8]. At the same time, modern on-bike sensors, such as power meters, GPS, and environmental sensors, can provide objective, real-time data to support more informed decisions.

Recent advances in the internet of things (IoT) enable interconnected sensing, data collection, and analytics in cycling [9]–[11]. Common sources include heart rate, cadence, power, and GPS-derived speed and elevation [12], [13], typically recorded by cyclo computers and synchronized to cloud platforms for post-ride analysis [14]–[16]. Despite this ecosystem, most work in IoT sports analytics emphasizes wearables or activity tracking rather than real-time multisensor fusion and decision support for tactical pacing [17]–[19].

Prior studies demonstrate the potential of machine learning for cycling-related prediction tasks, including real-time power estimation [20], race outcome forecasting [21], and broader health monitoring using IoT and deep learning [22]. Other contributions address discipline-specific performance analysis [23]

or embedded sensing devices such as TinyML-based power measurement [24]. While valuable, these efforts generally focus on single-target prediction, post-hoc analytics, or measurement systems, and do not provide an end-to-end solution that fuses multisource on-bike data and translates it into real-time, individualized speed recommendations. More broadly, existing IoT frameworks in sports and cycling are increasingly adopted, yet comprehensive real-time pacing guidance remains limited [25]–[30].

This gap motivates the present study. We propose an IoT-based framework that integrates multisensor data from a road bike and produces adaptive optimal speed recommendations during riding, accounting for both rider performance and external conditions [11], [31], [32]. The framework integrates IoT communication, edge computing, and machine learning, employing gradient boosting regression (GBR) to support real-time analysis and actionable pacing guidance. The main objective is to enable data-driven racing strategies that are practical for athletes and coaches in real-world road cycling.

2. METHOD

2.1. Proposed framework architecture

This work introduces an end-to-end architecture for predicting a cyclist's optimal speed from physiological and performance data, using machine learning to process these data and deliver real-time pacing guidance. The system is intended for direct use during training and competition. Using commercially available cycling sensors and lightweight edge devices (e.g., a laptop or a Raspberry Pi), without requiring constant cloud connectivity.

The architecture is organized into four layers: i) the data acquisition layer, which collects sensor data in real time; ii) the data processing and feature layer, which cleans, synchronizes, and engineers features from the raw data; iii) the predictive modeling layer, which uses machine learning to estimate an optimal speed; and iv) the recommendation and output layer, which converts the model output into pacing guidance that can be acted upon by the rider. Figure 1 shows this layered architecture and how information flows from sensing to recommendation. This layered structure supports modular development and practical deployment. Each layer can be maintained and extended independently. But all layers operate together to transform sensor and profile data into field-ready, personalized pacing recommendations.

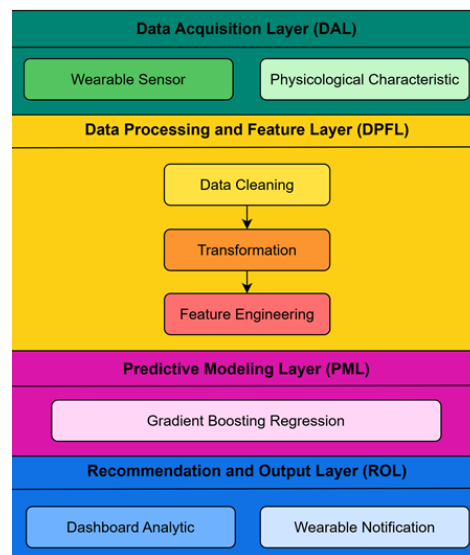


Figure 1. Proposed framework architecture

2.2. Framework architecture design

The architectural design ensures that the system can: i) ingest data from cycling sensors in realistic outdoor conditions, ii) process and interpret that data in near real time, iii) generate accurate predictions of optimal speed, and iv) present those predictions in a way that is understandable under physical load. Each layer plays a specific role in this pipeline. It is designed for practical use by cyclists.

2.2.1. Data acquisition layer

The data acquisition layer is responsible for collecting synchronized input from multiple sources during cycling. Data are captured via ANT+ and bluetooth low energy from: heart rate sensors (for training intensity and VO₂Max estimation [33]–[35]); power meters (for power in watts, used to derive functional threshold power (FTP), training stress score (TSS), and power-to-weight ratio [36]–[38]); cadence sensors (for pedaling rate in revolutions per minute (RPM) [39], [40]); and GPS (for speed, elevation, and trajectory [41], [42]). Physiological attributes such as age, weight, and height are also included, either manually or derived from the rider profile. The primary collector is a cycling head unit (e.g., Garmin Edge, Wahoo, XOSS), which records the ride and later exports data in formats such as .FIT, .TCX, or .CSV. Figure 2 illustrates the data acquisition layer data flow: sensors on the rider and bicycle send data to the cyclocomputer, which aggregates them and makes them available to the rest of the system pipeline.

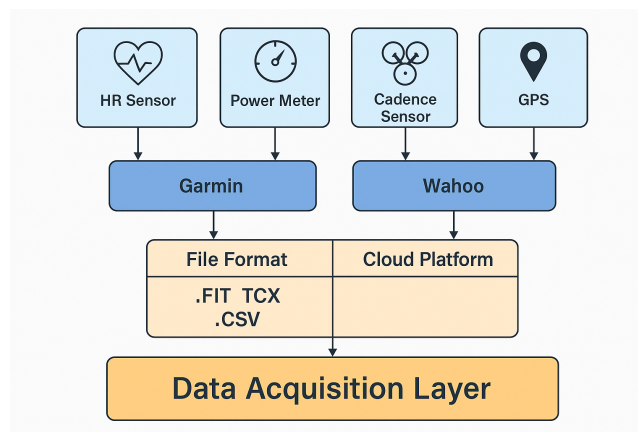


Figure 2. Data acquisition layer diagram

2.2.2. Data processing and feature layer

The data processing and feature layer transforms raw multi-sensor ride data into a model-ready dataset. First, streams with different sampling rates (heart rate, power, cadence, and GPS-derived elevation and speed) are aligned to a common time scale. Based on our earlier field trials, the inter-sensor delay was below 500 ms, and the processing loop averaged about 1.2 s, which is sufficient for soft real-time guidance but may still lag during abrupt transitions [43]. Next, missing values are imputed and noise or outliers are filtered. Then, signals are normalized (e.g., Min–Max scaling or Z-score standardization) to make variables with different units numerically comparable. Finally, domain features are engineered to capture physiological context, including FTP, TSS, estimated VO₂Max, and power-to-weight ratio (W/kg). The output is a structured multivariate dataset where each row corresponds to a time step in the ride, and each column represents a meaningful predictor for pacing.

2.2.3. Predictive modeling layer

The predictive modeling layer estimates the rider's optimal speed under current conditions. The main model is GBR [44]–[46], implemented in scikit-learn. We selected GBR to support semi-real-time inference on edge devices while remaining robust to outdoor sensing imperfections. In our earlier study on IoT-based road cycling analytics [43], GBR demonstrated low computational cost and good performance with smaller datasets (approximately 2,000 samples). In contrast, long short-term memory (LSTM) typically required larger data (at least 10,000 samples). The same prior work also indicated that GBR is easier to deploy and achieves lower edge latency and power consumption than LSTM in an ESP32 and Raspberry Pi edge cloud setup. Therefore, GBR provides a practical trade-off between accuracy and efficiency for responsive pacing recommendations, whereas LSTM can be explored for cloud-level temporal modeling in future work. Input features include heart rate, cadence, power, speed, elevation, FTP, VO₂Max estimate, TSS, and power-to-weight ratio. The prediction target is optimal speed: the sustainable speed that the rider can theoretically maintain under current physiological and terrain conditions.

Before training, the dataset is split (typically 80% for training / 20% for testing), and cross-validation is used to improve generalization. Performance is evaluated with mean absolute error (MAE), root mean square error (RMSE), and the coefficient of determination (R^2), which quantify absolute deviation, sensitivity to significant errors, and explained variance. Hyperparameters used to balance accuracy and robustness while keeping inference lightweight enough for on-device use.

2.2.4. Recommendation and output layer

The recommendation and output layer converts the predicted optimal speed into actionable pacing guidance. After the predictive modeling layer estimates the target speed, the recommendation and output layer contextualizes it using the current heart rate zone, cadence, elevation profile, power output, and fatigue load, then issues concise instructions such as hold pace, increase effort on flat, or reduce effort on climb. Guidance is also summarized through pacing zones (green = on target, yellow = underpaced, red = overpaced). Figure 3 presents Tkinter-based desktop interface that visualizes current versus optimal speed, pacing zone indicator, and time-series trends (e.g., heart rate and power), primarily for post-ride review and controlled training.

For in-ride use, the same recommendations can be delivered to a cycling head unit without causing significant distraction. Figure 4 shows a simplified wearable-style display (e.g., Garmin/Wahoo) presenting only essential feedback, namely current speed, target speed, and the pacing zone indicator. Reliable data transfer and recommendations are required across sensing, inference, and visualization. Figure 5 summarizes the communication flow from wearable sensors to local inference and output interfaces, while also supporting optional cloud synchronization through local transfer mechanisms (e.g., Bluetooth low energy or file synchronization).

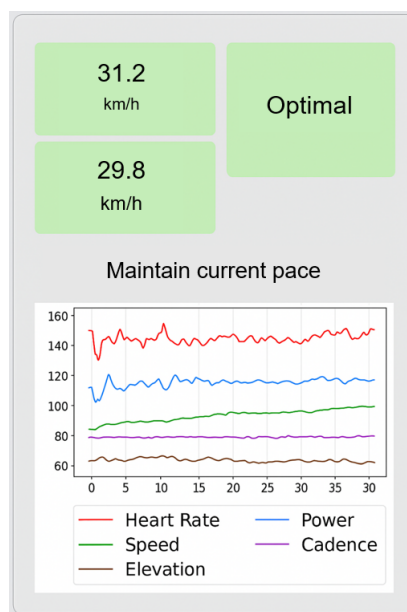


Figure 3. Tkinter-based visual interface



Figure 4. Wearable device display simulation (e.g., Garmin/Wahoo)

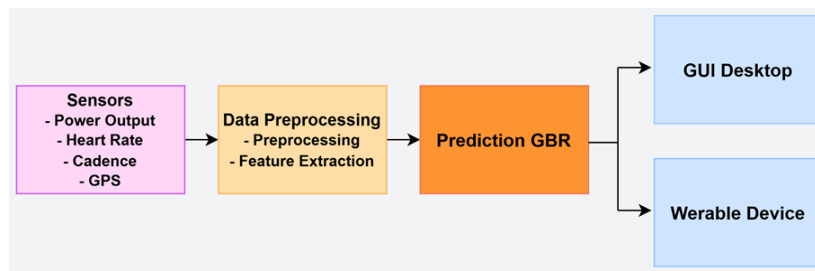


Figure 5. Data communication flow between sensing, inference, and visualization

2.3. Data acquisition and dataset collection

Training and evaluation data were collected from real cycling sessions using commercially available head units (Garmin Edge, Wahoo Elemnt, and XOSS) connected to heart rate, power, and cadence sensors via ANT+ or BLE [47], [48]. Activity logs were exported in standard formats such as .FIT, .TCX, and .CSV. Rider attributes (age, weight, height) and performance indicators (FTP and estimated VO₂Max) were recorded to support personalized modeling. As an example, the file 123816115880.FIT stores per-second records including heart rate, cadence, power, speed, elevation, and GPS coordinates. These records were parsed in Python using fitparse, converted into a pandas DataFrame, cleaned, and exported to .CSV. Table 1 shows a sample of the extracted second-by-second data.

More than 20 sessions (45–180 minutes each) were collected from multiple riders with informal consent. Preprocessing included timestamp alignment across sensors, outlier filtering, interpolation of missing samples, unit normalization, and tabular restructuring. Rider-level context (age, weight, height, FTP, and VO₂Max) was then merged into each second of activity data.

Table 2 summarizes the final dataset. Each row corresponds to one time step, and optimal speed is defined as the sustainable speed observed during efficient performance conditions, such as power near FTP and heart rate near the aerobic threshold. The optimal speed column serves as the supervised learning target.

Table 1. Example of extracted data from .FIT files

Timestamp	Lat	Long	Altitude	HR	Cadence	Distance	Speed	Power
2024-07-20T16:59:32	3.530037	98.67173	45.6	110	46	11.35	13.234	204
2024-07-20T16:59:33	3.530020	98.67176	45.6	110	55	14.32	15.685	182
2024-07-20T16:59:34	3.530012	98.67178	45.6	110	55	17.29	19.080	120
2024-07-20T16:59:35	3.530012	98.67182	45.6	112	60	21.57	19.314	120
2024-07-20T16:59:36	3.530009	98.67186	45.6	114	60	26.02	19.584	98
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 2. Final dataset structure after preprocessing

HR (bpm)	Cadence (RPM)	Power (W)	Elevasi (m)	Speed (km/h)	Age	Weight (kg)	Optimal speed (km/h)	Height (cm)	FTP (Watt)	VO ₂ Max
158	92	230	61.2	29.3	29	67	30.1	169	210	51
162	95	240	60.8	30.2	29	67	30.9	169	210	51
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

2.4. Predictive model development

The predictive model is implemented using GBR in scikit-learn [49], [50]. Each row of the dataset (Table 2) is treated as one supervised training instance, and the target is the optimal speed at that moment. The model is trained on 80% of the data and evaluated on the remaining 20%, with MAE, RMSE, and R² used as evaluation metrics [51], [52]. MAE measures average absolute error in km/h; RMSE penalizes large deviations more strongly; and R² measures explained variance. Hyperparameters such as n_estimators =100, learning_rate =0.1, and max_depth =3 were chosen to balance predictive power, robustness to noisy signals, and computational efficiency. The final trained model can be serialized and executed on a laptop or Raspberry Pi with low latency, enabling semi-real-time use.

2.5. System implementation and recommendation visualization

The proposed system is implemented as a local automated pipeline. Activity data from a cycling head unit (e.g., Garmin Edge) is exported in .FIT format, parsed in Python using `fitparse`, synchronized across sensor channels, cleaned, and converted to SI units. Key features are computed automatically, including FTP, estimated VO_2Max , TSS, gradient, and power-to-weight ratio. The resulting feature vector is then fed into the trained GBR model to estimate the optimal speed at each time step. Recommendations are presented through two interfaces.

Figure 3 shows a desktop visualization for post-ride review and controlled training, displaying current versus optimal speed together with heart rate, power, cadence, and pacing zone classification. For in-ride decision-making, Figure 4 presents a simplified on-device view that includes only essential feedback: current speed, target speed, and a color-coded pacing status. The end-to-end communication from sensing devices to local inference and user interfaces is summarized in Figure 5, including both local transfer (e.g., bluetooth low energy or file synchronization) and optional cloud synchronization (e.g., Garmin Connect). Overall, the pipeline converts raw sensor signals into clear, context-aware pacing guidance that can be acted on immediately in the field.

2.6. Project objectives and scope

This project implements and validates an IoT-based, real-time optimal pace prediction system for cyclists. The goal is to verify that the entire pipeline, from data acquisition to real-time pacing recommendations, can operate under realistic conditions and provide actionable guidance during training or racing. The system fuses data from wearable sensors (heart rate, power, cadence, GPS) and rider attributes (e.g., weight, age) to derive performance features, including FTP, estimated VO_2Max , TSS, and power-to-weight ratio. These features are fed into predictive models (GBR or LSTM) to estimate the optimal pace, which is then delivered via a desktop or wearable interface using color-coded pacing zones and tactical cues. The scope of this work includes offline (from .FIT files) and semi-real-time evaluation on constrained hardware (e.g., a laptop or Raspberry Pi), ensuring that the solution is technically feasible, practically deployable for athletes, and suitable as a basis for further system-level performance analysis.

2.7. Development platform and environment

The system was designed as an integrated hardware–software stack optimized for sensor interoperability, real-time inference, and in-field usability.

2.7.1. Hardware platform

The hardware architecture consists of wearable sensors and edge computing devices. A Garmin Edge 530/830 cyclocomputer acts as the central collector, receiving data from power meters, heart rate straps, cadence sensors, and GPS via ANT+ and bluetooth low energy, and recording them in .FIT format at one-second resolution. These data streams (power in watts for FTP/TSS and power-to-weight; heart rate for intensity and VO_2Max estimation; cadence for pedaling efficiency; GPS/elevation/speed for terrain context) are then exported for processing. Two edge compute targets are supported: i) a laptop (Intel Core i5/8 GB RAM / SSD) for model training, inference, visualization, and debugging; and ii) a Raspberry Pi 4 (4 GB RAM) for lightweight, local deployment capable of parsing .FIT files, extracting features, running the model, and driving a display via HDMI or a local interface. This modular layout separates sensing (wearables + Garmin Edge) from local analytics (laptop/Raspberry Pi), enabling autonomous operation without reliance on the cloud and making the system suitable for coaches and athletes in the field. Figure 6 illustrates this flow from sensors to prediction delivery.

2.7.2. Software platform

The software stack is implemented primarily in Python 3.10+ for portability, rapid data handling, and embedded machine learning support. Sensor data are ingested from .FIT files using `fitparse` and converted into pandas DataFrames, followed by cleaning, interpolation of missing values, time synchronization across channels, and normalization using pandas and numpy. Feature engineering then derives FTP, TSS, power-to-weight ratio, and estimated VO_2Max . The predictive core uses GBR via `scikit-learn`; trained models are serialized with `joblib` (.pkl) and evaluated with MAE, RMSE, and R^2 from `sklearn.metrics`. A Tkinter-based interface presents live/parsed outputs: actual vs. optimal pace, pacing zones (green/yellow/red), and heart rate/power trend plots rendered with `matplotlib`. The system runs offline on Windows or Raspbian and supports

semi-real-time replay of activity data; inference latency on the Raspberry Pi remains under 300 ms. Historical predictions and rider parameters are stored locally in SQLite to enable longitudinal analysis. This modular, open-source design is compatible with future extensions such as LSTM-based models, mobile deployment, or edge/cloud containers.

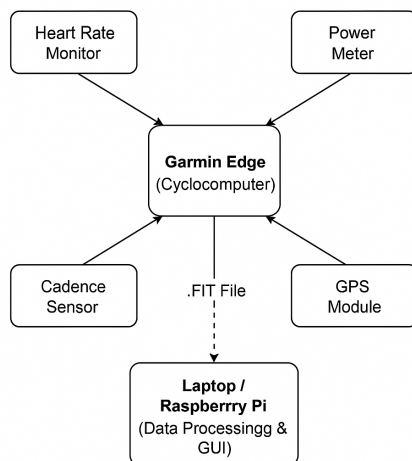


Figure 6. Hardware integration schematic

2.8. Architectural implementation

The overall system is implemented as a four-layer modular architecture: i) the data acquisition layer collects physiological and performance signals from sensors and rider context; ii) the data processing and feature layer cleans, synchronizes, and transforms raw signals into physiologically meaningful features; iii) the predictive modeling layer runs a personalized machine learning model (e.g., GBR or LSTM) to estimate optimal pace in context; and iv) the recommendation and output layer converts predicted pace into actionable guidance via pacing zones and alerts. The novelty of the system lies in integrating terrain-aware and physiology-aware features with adaptive prediction and immediately usable recommendations, all running locally on a laptop or Raspberry Pi without requiring a remote server. Practical testing indicates that this architecture is responsive, stable, and suitable for field use, supporting data-driven pacing decisions rather than relying solely on subjective judgment.

2.9. End-to-end system integration process

The full workflow is designed to run automatically from input to recommendation with minimal manual intervention. During a ride, the Garmin Edge records heart rate, power, cadence, speed, elevation, and GPS position and exports them as a .FIT file. That file is then parsed using fitparse into structured data, cleaned, time-aligned, and enriched with engineered features such as FTP, TSS, VO_2 Max estimation, and gradient. The resulting feature stream is fed into the trained predictive model (e.g., GBR), which infers the optimal speed for each time step locally on either the laptop or the Raspberry Pi. The final output is presented via a Tkinter interface that displays actual vs. optimal speed, pacing zone classification (green = on target, yellow = underpaced, red = overpaced), route-relevant suggestions, and physiological trends (heart rate and power curves). The entire pipeline can be executed by providing a .FIT file or selecting a recorded activity, after which processing, inference, and visualization proceed automatically. Figure 7 summarizes this end-to-end integration.

2.10. Real-world implementation case study

A real-world case study was conducted to validate the system during an outdoor ride. An experienced cyclist completed a 42 km route with approximately 420 m elevation gain (flats, climbs, and descents) in about 1 hour 50 minutes using a Garmin Edge 530 paired with power, heart rate, and cadence sensors. The recorded .FIT file was processed end-to-end through parsing, cleaning, interpolation of missing samples, timestamp synchronization, and feature generation (e.g., FTP, power-to-weight ratio, and local gradient).

The resulting features were evaluated using a locally deployed GBR model (on a laptop or Raspberry Pi), yielding time-step-optimal speed estimates with inference latency below 100 ms per data point.

The predicted optimal speed was then translated into adaptive pacing guidance. For sustained climbs above 5% gradient, the system recommended reducing effort to remain within a sustainable physiological zone; for flatter segments, it suggested increasing speed while maintaining a cadence of 85–95 RPM. The visualization also summarized pacing adherence, showing that the rider spent 68% of the session in the optimal (green) zone, 24% underpaced (yellow), and 8% overpaced (red), alongside heart rate and power trends for rider and coach interpretation. Figure 8 links the elevation profile with the pacing-zone timeline, while Figure 9 compares actual speed against the model-recommended optimal speed. Overall, the case study indicates that the pipeline is stable for outdoor use, the feedback is interpretable for managing effort and fatigue, and the workflow remains compatible with common training ecosystems such as Garmin Connect, Strava, and TrainingPeaks, supporting future deployment for athlete training and wearable coaching.

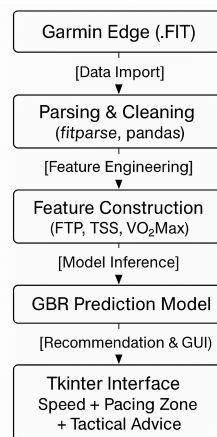


Figure 7. End-to-end system integration diagram

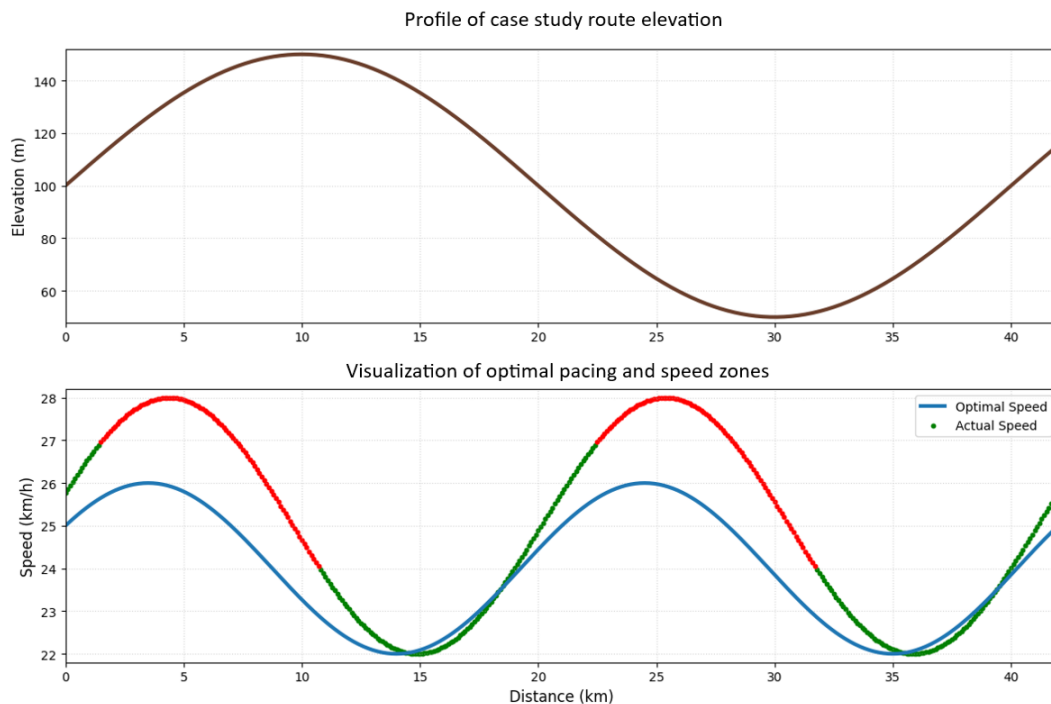


Figure 8. Route elevation and pacing zone visualization

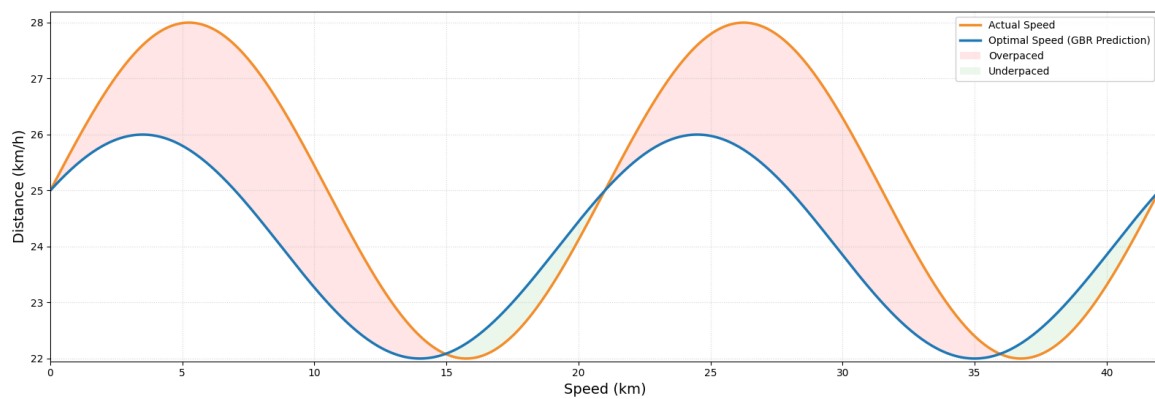


Figure 9. Actual speed vs. optimal speed comparison graph

3. RESULTS AND DISCUSSION

3.1. Evaluation preparation

The evaluation was designed to be systematic, objective, and representative of real-world cycling. Three aspects were assessed: the prediction accuracy of the GBR model for optimal speed; the reliability of the IoT pipeline for multisensor acquisition and synchronization; and the practical usefulness of pacing-zone recommendations during riding. Model performance was quantified using MAE, RMSE, and R^2 . System reliability was evaluated through latency, synchronization consistency across sensor channels, and end-to-end response time.

The test data comprised real-world cycling sessions over varied terrain, including physiological signals, GPS-derived position and elevation, and observed rider speed. The system was deployed under realistic conditions, and its predicted optimal speed was compared against the observed ride profile. In addition to quantitative metrics, usability was assessed through rider interaction with both the wearable-style display and the desktop dashboard, supported by structured observations and short interviews focusing on feedback clarity, pacing guidance, and perceived effort regulation. This preparation ensured the system was evaluated not only for numerical accuracy and technical robustness, but also for practical usability in the field.

3.2. System evaluation dimensions

The system was evaluated across four key dimensions to verify its real-world viability for cycling: i) prediction accuracy of the GBR model for optimal speed, ii) reliability of the end-to-end IoT pipeline in acquiring, synchronizing, transmitting, and processing multisensor data in real time, iii) effectiveness of pacing zone guidance in shaping rider effort, and iv) user experience in terms of clarity, usability, and influence on pacing decisions. This multidimensional evaluation combines quantitative metrics, system-level performance, and rider feedback to confirm not only that the model produces reasonable outputs. But also that those outputs can be delivered with low latency, acted upon in real-world conditions, and understood by cyclists under physical load.

3.2.1. Prediction model accuracy evaluation

The GBR model was selected to generate optimal speed recommendations based on features including heart rate, gradient, elevation, and recent speed. A time-series split was used to avoid leakage between training and test data. Accuracy was evaluated on real-world cycling data using MAE, RMSE, and R^2 . Results are reported in Table 3: MAE of 7.12, RMSE of 9.24, and $R^2 = 0.813$, meaning the model explains about 81.3% of the variation in optimal speed. Although the absolute error (on the order of $\pm 7-9$ km/h) is not negligible, it is acceptable because the system provides guidance in pacing zones (optimal / underpaced / overpaced) rather than demanding an exact numeric target. As shown in Figure 10, the predicted optimal speed tracks actual speed trends well on flats and climbs, with larger deviations during steep descents, where factors such as braking and wind are not yet modeled. This level of performance is sufficient for real-time pacing support and can be improved with additional context features.

Table 3. Model performance evaluation results on test data

Evaluation metric	Result
MAE	7.12
RMSE	9.24
R ² score	0.813

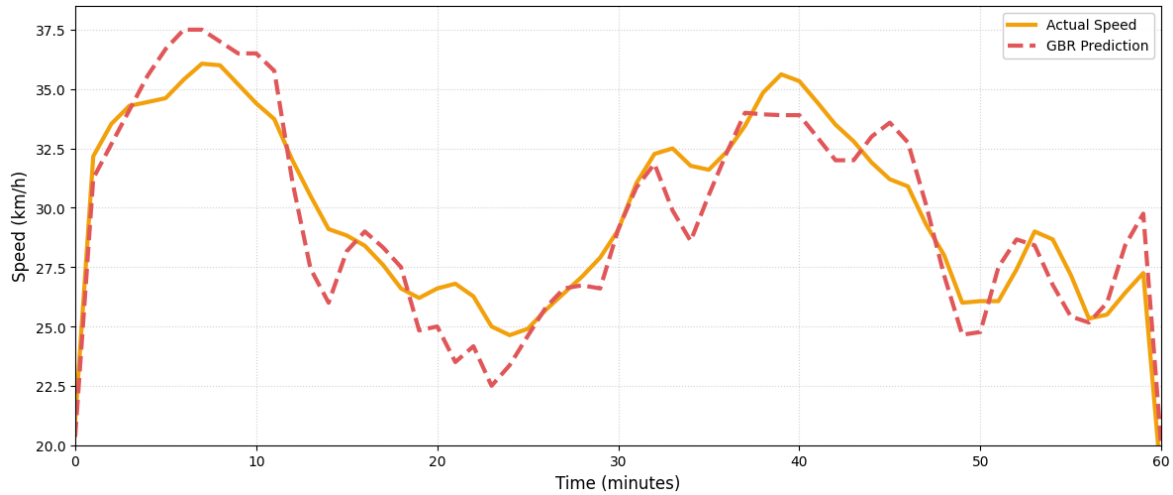


Figure 10. Comparison between actual speed and GBR-predicted optimal speed

To contextualize the reported GBR scores without introducing additional tables, we relate our findings to prior literature and our earlier work. Several cycling analytics studies employ statistical or linear formulations, which are often limited when the relationships among physiological load, terrain, and speed are strongly nonlinear [9], [6]. In our prior benchmarking study on IoT-based road cycling analytics, we evaluated lightweight baselines, such as linear regression, random forests, and autoregressive integrated moving average (ARIMA), alongside more advanced models. We found that GBR consistently outperformed these baselines while remaining computationally efficient for practical IoT deployment [43]. Although LSTM achieved the highest predictive accuracy in that earlier study, it required larger datasets and higher computational resources. In contrast, GBR offered a more favorable trade-off between accuracy and efficiency for low-latency edge inference [43]. These results support the use of GBR in the present framework and explain why it outperforms purely linear baselines while maintaining system responsiveness for training guidance.

3.2.2. End-to-end internet of things system reliability evaluation

System reliability was assessed across sensing, communication, processing, and field durability. Multisensor data (heart rate, wheel speed, GPS, rider input) remained synchronized with an average inter-sensor delay <500 ms, showed no significant missing values across 10 outdoor sessions, and did not require restarts. Wireless transmission latency was 200–350 ms per cycle with <2% data loss within a 10 m range, and the system automatically recovered after brief link drops. The whole processing loop averaged 1.2 s, with visualization updates occurring within 300 ms of model output, satisfying soft real-time requirements for cycling. Hardware and interface also remained functional under vibration, motion, and splash exposure in the field. These results indicate that the architecture is sufficiently stable for live training, not only under laboratory conditions.

3.2.3. Effectiveness of pacing zone recommendations

To turn numerical predictions into actionable guidance, the system continuously classifies the rider's current effort into three pacing zones: green (within $\pm 5\%$ of the optimal range), yellow (underpaced), and red (overpaced). These zones are displayed in real time on both wearable-style and desktop interfaces. Figure 11 and Table 4 show that riders remained in the green (optimal) zone for 58.4% of total duration, with 24.6% in yellow (too easy) and 17.0% in red (too hard). The mean deviation in each zone indicates how far the rider was

from the model’s target. The fact that deviations outside the green zone were typically short-lived suggests that riders responded to feedback and corrected their pacing during the session. This confirms that the system not only predicts but also actively supports in-ride pacing control.

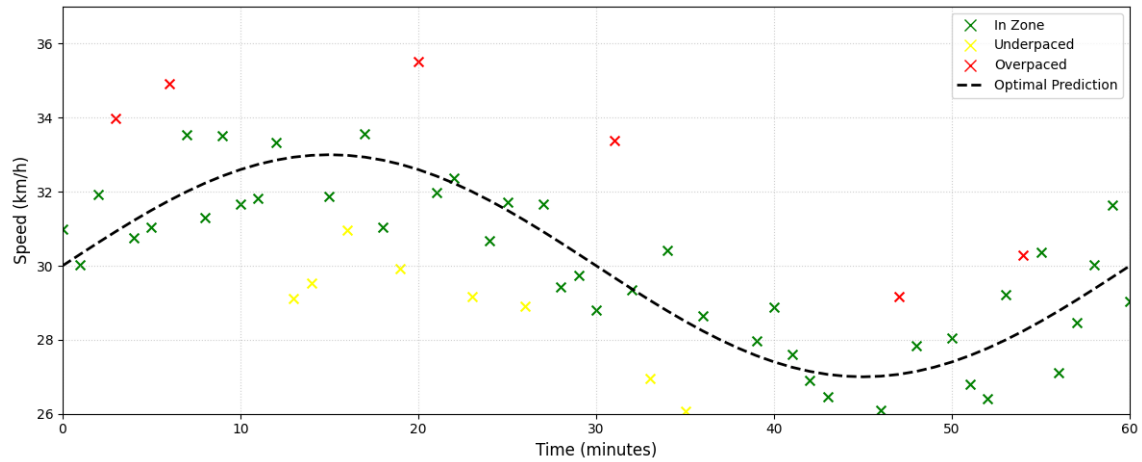


Figure 11. Distribution of actual speed points by pacing zone

Table 4. Pacing zone distribution

Pacing zone	Proportion of time (%)	Mean speed deviation
Green	58.4	±2.1 km/h
Yellow	24.6	-6.7 km/h
Red	17.0	+7.8 km/h

3.2.4. User perceptions and experience

Ten cyclists tested the whole system (sensors, predictive model, pacing zone interface) in real outdoor rides. Most participants described the interface as easy or very easy to interpret, and the guidance as relevant, particularly for climbs or high-intensity segments. Riders reported that the color-zoned pacing feedback improved effort awareness, reduced overpacing in the red zone, and encouraged them to increase effort when drifting into the yellow zone. They also stated that the system helped them become more disciplined and more aware of their physical limits than relying solely on “feel”. Table 5 summarizes individual feedback, including perceived ease of use, relevance, effect on pacing strategy, and requested improvements. Common suggestions included vibration or audio alerts, handlebar LEDs, post-session summaries of pacing history, and customizable training modes (endurance, interval, race simulation). Overall, user feedback supports the system’s practical value and readiness for real-world cycling applications.

Table 5. Summary of user perceptions and experiences

No	Initials	Ease	Relevance	Influence strategy	Suggestion
1	P01	Very easy	Very relevant	Adjusting incline	Add vibration
2	P02	Easy	Relevant	Holding the start	Minimalist display
3	P03	Very easy	Very relevant	Speed down	Post-workout summary
4	P04	Easy	Somewhat relevant	Performance awareness	App integration
5	P05	Easy	Relevant	Pace discipline	Endurance/interval mode
6	P06	Very easy	Very relevant	Red zone response	Voice notification
7	P07	Easy	Relevant	Terrain adaptation	Competition mode
8	P08	Easy	Relevant	Interval correction	LED on handlebar
9	P09	Very easy	Very relevant	Aware of physical limits	Adjust zone sensitivity
10	P10	Easy	Somewhat relevant	Maintain pace	Cumulative visualization

3.3. Limitations and development implications

A system evaluation is incomplete without an assessment of existing limitations. Identifying limitations helps improve system quality and provides a basis for further development. This evaluation also

provides an objective picture of the system's reliability and applicability. By understanding the limitations, improvement strategies can be formulated more effectively.

One major limitation is the limited number and diversity of users in the trial. The ten participants were from recreational and semi-competitive categories. There was no representation of elite athletes, seniors, or users with special needs. A complete participant profile is shown in Table 6.

Table 6. System test respondent profile

No.	Initials	User type	Fitness	Cycling experience	Tech experience	Notes
1	P01	Recreational	Moderate	> 2 years	Advanced	-
2	P02	Amateur (training)	High	> 3 years	Advanced	-
3	P03	Beginner	Low	< 1 year	Moderate	-
4	P04	Recreational	Moderate	1–2 years	Limited	-
5	P05	Competitive training	High	> 4 years	Advanced	-
6	P06	Beginner	Low	< 1 year	Limited	UI navigation difficulties
7	P07	Recreational	Moderate	2–3 years	Moderate	-
8	P08	Interval training	High	> 5 years	Advanced	-
9	P09	Recreational	Moderate	> 1 year	Moderate	-
10	P10	Structured training	High	> 4 years	Advanced	-

Another limitation lies in the static prediction model. The system is not yet able to adjust predictions in response to changes in user performance over time. In some cases, model accuracy decreased after several days of use. This indicates the need for personalization or continuous learning.

From a hardware perspective, the wearable's battery life remains suboptimal for cycling activities lasting more than 3 hours. Furthermore, data dropout occurred under low-signal conditions. A summary of the system's technical performance during testing is presented in Table 7. The system also lacks multimodal feedback, such as vibration or sound, a feature many users desire. Requests for additional features during the evaluation are summarized in Table 8.

Table 7. System operational performance during testing

Parameters	Average	Notes
Battery life	2 hours 45 minutes	Not optimal for >3 hours
Sensor data dropout	1.7%	Occurs when the signal is weak
UI latency	280 ms	Increases when the connection is unstable
Prediction processing time	1.2 seconds	Stable

Table 8. Additional feature requests by respondents (n =10)

Requested features	Number of respondents	Percentage (%)
Vibration when zone changes	6	60
Post-activity pacing zone summary	5	50
Personalized training mode	4	40
Audio notifications	3	30
Integration with third-party apps	4	40

The data above indicate that the development of interactive features and personalization remains highly needed. The system also does not yet support open integration with popular apps, such as Garmin, Strava, or TrainingPeaks. This is a barrier to system adoption in the broader digital training ecosystem.

Based on the evaluation, five main development directions can be formulated:

- Implementation of adaptive models and personalization based on historical user data.
- Integration with cloud computing for long-term training and online collaboration.
- Development of wearables with haptic/audio support and improved power efficiency.
- Implementation of open APIs for integration across training apps.
- Improved multimodal interaction for faster response in various conditions.

By understanding these limitations in a data-driven manner, future system development can be more focused, adaptive, and tailored to a broader range of user needs.

4. CONCLUSIONS

This study presents a deployable IoT-based framework for road bike speed optimization that goes beyond offline speed prediction by integrating multisensor data acquisition, physiologically informed feature engineering, terrain-aware machine learning, and real-time pacing guidance in a single end-to-end pipeline. The main novelty of this work lies in combining physiological, performance, and route-context variables within an adaptive prediction-and-recommendation architecture that can operate locally on edge devices, enabling practical in-ride decision support without relying on a remote server. The experimental results confirm that the proposed framework is both accurate and operationally feasible. The GBR model achieved an MAE of 7.12, an RMSE of 9.24, and an R^2 of 0.813, while the system maintained soft real-time performance with an average processing cycle of about 1.2 s and inference latency below 300 ms on Raspberry Pi. In practice, riders remained in the green pacing zone for 58.4% of total riding time, indicating that the framework provides actionable pacing support rather than merely producing numerical predictions. The principal contribution of this research is therefore not only a predictive model but also a validated, modular system architecture for personalized pacing assistance in road cycling, suitable for further real-world deployment. Although the current evaluation is limited to a relatively small set of riders, routes, and device conditions, and external factors such as wind and braking are not yet explicitly modeled, the findings demonstrate the potential of edge-based IoT intelligence to support more disciplined, data-driven pacing strategies. Future work should focus on broader validation, richer contextual sensing, adaptive personalization, and integration with wearable and coaching ecosystems to extend the framework toward fully real-time training and competitive applications.

FUNDING INFORMATION

The authors acknowledge that this research was financially supported by Universitas Sumatera Utara.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Tigor Hamonangan Nasution	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	✓
Opim Salim Sitompul	✓			✓						✓		✓		
Fahmi Fahmi	✓			✓						✓		✓		
Muhammad Anggia Muchtar	✓			✓						✓		✓		

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal Analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project Administration

Fu : Funding Acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The authors also declare that they have no non-financial competing interests, including political, personal, religious, ideological, academic, or intellectual interests, that could be perceived as influencing this work.

DATA AVAILABILITY

The data that support the findings of this study are available on request from the corresponding author, [THN]. The data, which contain information that could compromise the privacy of research participants, are not publicly available due to certain restrictions.





REFERENCES

- [1] M.-P. Veillette, E. Grisé, and A. El-Geneidy, "Does one bicycle facility type fit all? evaluating the stated usage of different types of bicycle facilities among cyclists in Quebec City, Canada," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2673, no. 6, pp. 650–663, Jun. 2019, doi: 10.1177/0361198119844741.
- [2] J. Viellehner and W. Potthast, "The effect of road-bike damping on neuromuscular short-term performance," *Sports Biomechanics*, vol. 19, no. 6, pp. 723–737, Nov. 2020, doi: 10.1080/14763141.2020.1797153.
- [3] T. H. Nasution, A. Sitohang, - Seniman, and - Soeharwinto, "Bike fitting system based on digital image processing on road bike," *JOIV : International Journal on Informatics Visualization*, vol. 9, no. 2, Mar. 2025, doi: 10.62527/joiv.9.2.2796.
- [4] D. Sanders and T. V. Erp, "The physical demands and power profile of professional men's cycling races: an updated review," *International Journal of Sports Physiology and Performance*, vol. 16, no. 1, pp. 3–12, Jan. 2021, doi: 10.1123/ijspp.2020-0508.
- [5] D. G. Wilson and T. Schmidt, *Bicycling Science*. Cambridge, United States: The MIT Press, 2020, doi: 10.7551/mitpress/11660.001.0001.
- [6] A.-W. D. Leeuw, M. Heijboer, T. Verdonck, A. Knobbe, and S. Latré, "Exploiting sensor data in professional road cycling: personalized data-driven approach for frequent fitness monitoring," *Data Mining and Knowledge Discovery*, vol. 37, no. 3, pp. 1125–1153, May 2023, doi: 10.1007/s10618-022-00905-5.
- [7] M. W. O'Grady, R. Worn, J. O. Owens, B. J. O'Brien, and S. W. Talpey, "Race craft: A qualitative exploration of the development, implementation and reflection of tactical decision making in road cycling," *International Journal of Sports Science & Coaching*, vol. 18, no. 6, pp. 2160–2170, Dec. 2023, doi: 10.1177/17479541221118423.
- [8] E. Kolar, R. Biloslavo, R. Pišot, S. Veličković, and M. Tušak, "Conceptual framework of coaches' decision-making in conventional sports," *Frontiers in Psychology*, vol. 15, Jan. 2025, doi: 10.3389/fpsyg.2024.1498186.
- [9] X. Qian, J. K. Moore, and D. Niemeier, "Predicting bicycle pavement ride quality: sensor-based statistical model," *Journal of Infrastructure Systems*, vol. 26, no. 3, Sep. 2020, doi: 10.1061/(ASCE)IS.1943-555X.0000571.
- [10] C.-H. Ho, J. Gao, M. Snyder, and P. Qiu, "Development and application of instrumented bicycle and its sensing technology in condition assessments for bike trails," *Journal of Infrastructure Systems*, vol. 27, no. 3, Sep. 2021, doi: 10.1061/(ASCE)IS.1943-555X.0000632.
- [11] G. Catargiu, E.-H. Dulf, and L. C. Miclea, "Connected bike-smart IoT-based cycling training solution," *Sensors*, vol. 20, no. 5, Mar. 2020, doi: 10.3390/s20051473.
- [12] T. Matyja, A. Kubik, and Z. Stanik, "Possibility to use professional bicycle computers for the scientific evaluation of electric bikes: velocity, cadence and power data," *Energies*, vol. 15, no. 3, Feb. 2022, doi: 10.3390/en15031127.
- [13] O. Siddiqui, S. DiBiase, R. Hoang, B. Nguyen, O. Khan, and N. Famiglietti, "Evaluating the accuracy and reliability of bicycle GPS devices," *SAE International Journal of Advances and Current Practices in Mobility*, vol. 03, no. 6, pp. 3093–3114, Apr. 2021, doi: 10.4271/2021-01-0882.
- [14] R. Venkatesan, H. Lopes, and G. Yemen, "Wahoo fitness: segmentation and data Insights," SSRN, 2022.
- [15] D. J. Rivers, "Strava as a discursive field of practice: Technological affordances and mediated cycling motivations," *Discourse, Context & Media*, vol. 34, Apr. 2020, doi: 10.1016/j.dcm.2019.100345.
- [16] H. H. Hochmair, E. Bardin, and A. Ahmouda, "Estimating bicycle trip volume for Miami-Dade county from Strava tracking data," *Journal of Transport Geography*, vol. 75, pp. 58–69, Feb. 2019, doi: 10.1016/j.jtrangeo.2019.01.013.
- [17] B. McIlroy, L. Passfield, H.-C. Holmberg, and B. Sperlich, "Virtual training of endurance cycling – a summary of strengths, weaknesses, opportunities and threats," *Frontiers in Sports and Active Living*, vol. 3, Mar. 2021, doi: 10.3389/fspor.2021.631101.
- [18] F. J. Dian, R. Vahidnia, and A. Rahmati, "Wearables and the internet of things (IoT), applications, opportunities, and challenges: a survey," *IEEE Access*, vol. 8, pp. 69200–69211, 2020, doi: 10.1109/ACCESS.2020.2986329.
- [19] M. Rana and V. Mittal, "Wearable sensors for real-time kinematics analysis in sports: a review," *IEEE Sensors Journal*, vol. 21, no. 2, pp. 1187–1207, Jan. 2021, doi: 10.1109/JSEN.2020.3019016.
- [20] Y. Kataoka and P. Gray, "Real-time power performance prediction in tour de France," in *Machine Learning and Data Mining for Sports Analytics: 5th International Workshop, MLSA 2018 Proceedings*, Dublin, Ireland, 2018, pp. 121–130, doi: 10.1007/978-3-030-17274-9_10.
- [21] L. Kholikine, T. D. Schepper, T. Verdonck, and S. Latré, "A machine learning approach for road cycling race performance prediction," in *Machine Learning and Data Mining for Sports Analytics (MLSA 2020)*, vol. 1324, 2020, pp. 103–112, doi: 10.1007/978-3-030-64912-8_9.
- [22] Z. Chao, L. Yi, L. Min, and Y. Y. Long, "IoT-enabled prediction model for health monitoring of college students in sports using big data analytics and convolutional neural network," *Mobile Networks and Applications*, Sep. 2024, doi: 10.1007/s11036-024-02370-4.
- [23] B. Ofoghi, J. Zeleznikow, C. MacMahon, and D. Dwyer, "A machine learning approach to predicting winning patterns in track cycling omnium," in *Artificial Intelligence in Theory and Practice III*, vol. 331, 2010, pp. 67–76, doi: 10.1007/978-3-642-15286-3_7.
- [24] V. Luder, S. Bian, and M. Magno, "CycloWatt: an affordable, TinyML-enhanced IoT device revolutionizing cycling power metrics," 2024 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Glasgow, United Kingdom, 2024, pp. 1–6, doi: 10.1109/I2MTC60896.2024.10560804.
- [25] M. Bhatia, "IoT-inspired framework for athlete performance assessment in smart sport industry," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9523–9530, Jun. 2021, doi: 10.1109/JIOT.2020.3012440.
- [26] Y. Qiu, G. Liu, B. A. Muthu, and C. B. Sivaparthipan, "Design of an energy-efficient IoT device with optimized data management in sports person health monitoring application," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 10, Oct. 2022, doi: 10.1002/ett.4258.
- [27] R. K. Navandar, S. H. Hasan, N. Jadhav, K. U. Singh, R. Monisha, and N. Venkatram, "Modernizing sports an intelligent strategy for entertainment through internet of things in sports," *Entertainment Computing*, vol. 52, Jan. 2025, doi: 10.1016/j.entcom.2024.100804.
- [28] L. E. Olmos *et al.*, "A data science framework for planning the growth of bicycle infrastructures," *Transportation Research Part C: Emerging Technologies*, vol. 115, Jun. 2020, doi: 10.1016/j.trc.2020.102640.
- [29] F. Oliveira and D. G. Costa, "Toward sustainable cycling: modelling and visualization issues of cycle paths for IoT-based sensing," in *2021 IEEE Globecom Workshops (GC Wkshps)*, Dec. 2021, pp. 1–6, doi: 10.1109/GCWkshps52748.2021.9682043.




- [30] F. Behrendt, "Mobility and data: cycling the utopian internet of things," *Mobilities*, vol. 15, no. 1, pp. 81–105, Jan. 2020, doi: 10.1080/17450101.2019.1698763.
- [31] S. M. Kumar, A. Mohanty, R. Raman, M. Muthulekshmi, and A. Barve, "Smart biking: IoT-connected cycling gear for training and safety," in *2023 Second International Conference On Smart Technologies For Smart Nation (SmartTechCon)*, Aug. 2023, pp. 652–656, doi: 10.1109/SmartTechCon57526.2023.10391382.
- [32] W. N. W. Muhamad, S. A. B. Razali, N. A. Wahab, M. M. Azreen, S. S. Sarnin, and N. F. Naim, "Smart bike monitoring system for cyclist via internet of things (IoT)," in *2020 IEEE 5th International Symposium on Telecommunication Technologies (ISTT)*, Nov. 2020, pp. 168–173, doi: 10.1109/ISTT50966.2020.9279377.
- [33] B. Carrier, M. M. Helm, K. Cruz, B. Barrios, and J. W. Navalta, "Validation of aerobic capacity (VO₂max) and lactate threshold in wearable technology for athletic populations," *Technologies*, vol. 11, no. 3, May 2023, doi: 10.3390/technologies11030071.
- [34] B. Carrier, S. M. Chaves, and J. W. Navalta, "Validation of aerobic capacity (VO₂max) and pulse oximetry in wearable technology," *Sensors*, vol. 25, no. 1, Jan. 2025, doi: 10.3390/s25010275.
- [35] R. Lambe, B. O'Grady, M. Baldwin, and C. Doherty, "Investigating the accuracy of Apple Watch VO₂ max measurements: a validation study," *PLoS One*, vol. 20, no. 5, May 2025, doi: 10.1371/journal.pone.0323741.
- [36] F. K. Borszcz, A. F. Tramontin, and V. P. Costa, "Reliability of the functional threshold power in competitive cyclists," *International Journal of Sports Medicine*, vol. 41, no. 3, pp. 175–181, Mar. 2020, doi: 10.1055/a-1018-1965.
- [37] G. Gallo *et al.*, "Relationships between training dose and record power outputs in professional road cyclists: insights and threats to validity," *Biology of Sport*, vol. 40, no. 2, pp. 485–495, 2023, doi: 10.5114/biolsport.2023.114284.
- [38] M. M. March *et al.*, "Power profile index: An adjustable metric for load monitoring in road cycling," *Applied Sciences*, vol. 12, no. 21, Oct. 2022, doi: 10.3390/app122111020.
- [39] H. Charvatova, A. Prochazka, O. Vysata, C. P. S. Araujo, and J. H. Smith, "Evaluation of accelerometric and cycling cadence data for motion monitoring," *IEEE Access*, vol. 9, pp. 129256–129263, 2021, doi: 10.1109/ACCESS.2021.3111323.
- [40] H. Mihaldinec, L. Klaić, and H. Džapo, "Estimation of the cycling cadence using low-cost inertial sensors mounted on a bicycle," in *2023 6th Conference on Cloud and Internet of Things (CIoT)*, Lisbon, Portugal, Mar. 2023, pp. 24–28, doi: 10.1109/CIoT57267.2023.10084907.
- [41] E. Berjisian, A. Bigazzi, and H. Barkh, "By cyclists, for cyclists: road grade and elevation estimation from crowd-sourced fitness application data," *PLoS ONE*, vol. 18, no. 12, Dec. 2023, doi: 10.1371/journal.pone.0295027.
- [42] M. Wang, J. Haworth, I. Ilyankou, and N. Christie, "CycleTrajectory: an end-to-end pipeline for enriching and analyzing GPS trajectories to understand cycling behavior and environment," in *Proceedings of the 2nd ACM SIGSPATIAL Workshop on Sustainable Urban Mobility*, Oct. 2024, pp. 27–32, doi: 10.1145/3681779.3696838.
- [43] T. H. Nasution, O. S. Sitompul, F. Fahmi, and M. A. Muchtar, "Comparative analysis of artificial intelligence algorithms for optimizing IoT and big data frameworks in road bike applications," *IAENG International Journal of Applied Mathematics*, vol. 56, no. 1, 2026.
- [44] S. Wassanet *et al.*, "Gradient boosting for health IoT Federated learning," *Sustainability*, vol. 14, no. 24, Dec. 2022, doi: 10.3390/su142416842.
- [45] C. Bentéjac, A. Csörgő, and G. M. Muñoz, "A comparative analysis of gradient boosting algorithms," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 1937–1967, Mar. 2021, doi: 10.1007/s10462-020-09896-5.
- [46] S. Sang, F. Qu, and P. Nie, "Ensembles of gradient boosting recurrent neural network for time series data prediction," *IEEE Access*, vol. 13, pp. 122331–122340, 2025, doi: 10.1109/ACCESS.2021.3082519.
- [47] N. Khuri, E. M. Burford, S. Parsons, and C. Xu, "A game-theoretical approach for data acquisition from fitness tracking devices," in *2021 International Symposium on Networks, Computers and Communications (ISNCC)*, Oct. 2021, pp. 1–6, doi: 10.1109/ISNCC52172.2021.9615729.
- [48] D. Sweet and G. Bretting, "FIT file processing for accident reconstruction," *WCX SAE World Congress Experience*, Apr. 2025, doi: 10.4271/2025-01-8696.
- [49] F. Nelli, "Machine learning with scikit-learn," in *Python Data Analytics*, Berkeley, California: Apress, 2023, pp. 259–287, doi: 10.1007/978-1-4842-9532-8.8.
- [50] M. Savic *et al.*, "The application of machine learning techniques in prediction of quality of life features for cancer patients," *Computer Science and Information Systems*, vol. 20, no. 1, pp. 381–404, 2023, doi: 10.2298/CSIS220227061S.
- [51] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *PeerJ Computer Science*, vol. 7, Jul. 2021, doi: 10.7717/peerj-cs.623.
- [52] F. Merabet and D. Benmerzoug, "QoS prediction for service selection and recommendation with a deep latent features autoencoder," *Computer Science and Information Systems*, vol. 19, no. 2, pp. 709–733, 2022, doi: 10.2298/CSIS210518054M.

BIOGRAPHIES OF AUTHORS






Tigor Hamonangan Nasution     is a lecturer in Electrical Engineering at the Universitas Sumatera Utara. He has been a lecturer since 2014. He is also a senior member of IEEE. He completed his master's in Electrical Engineering in 2014 and earned his Doctorate in Computer Science from Universitas Sumatera Utara in 2025. He is active in research in computer engineering, especially in embedded systems and artificial intelligence. In addition, he serves as an advisor to the student activity unit in robotics at Universitas Sumatera Utara. He can be contacted at email: tigor.nasution@usu.ac.id.






Opim Salim Sitompul    is a professor at the Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, and currently serves as vice rector IV of Universitas Sumatera Utara. His expertise lies in data structures, databases, and information systems, with active research focusing on business informatics and pattern recognition. He can be contacted at email: opimsalim@usu.ac.id.



Fahmi Fahmi    is a professor in Electrical Engineering at the Faculty of Engineering. He completed his doctoral degree in Biomedical Engineering at the Universiteit van Amsterdam, the Netherlands, in 2015. He was once the head of the Electrical Engineering Study Program, a member of the academic senate, and a member of the board of trustees at Universitas Sumatera Utara. Currently, he is the dean at Faculty of Engineering, Universitas Sumatera Utara, in addition to his primary role as a lecturer and researcher in electrical and biomedical engineering. He can be contacted at email: fahmimn@usu.ac.id.



Muhammad Anggia Muchtar    is a lecturer at the Faculty of Computer Science and Information Technology, Universitas Sumatera Utara. He completed his doctoral degree in 2021. From 2016 to 2022, he served as head of the Information Systems Center at Universitas Sumatera Utara and is currently serving as head of the Evaluation Board for Priority Programs at Universitas Sumatera Utara. His areas of expertise include natural language processing (NLP), multimedia, and software requirements engineering. He can be contacted at email: anggia.muchtar@usu.ac.id.