engineering proceedings

MDPI

Proceeding Paper

Prediction-Based Tip-Over Prevention for Planetary Exploration Rovers [†]

Siddhant Shete [1,*], Raúl Domínguez [1], Ravisankar Selvaraju [1], Mariela De Lucas Álvarez [1] and Frank Kirchner [1,2]

[1] Robotics Innovation Center of German Research Center for Artificial Intelligence GmbH (DFKI), 28359 Bremen, Germany; raul.dominguez@dfki.de (R.D.); ravisankar.selvaraju@dfki.de (R.S.); mariela.de_lucas_alvarez@dfki.de (M.D.L.Á.); frank.kirchner@dfki.de (F.K.)

[2] Robotics Research Group, University of Bremen, 28359 Bremen, Germany

[*] Correspondence: siddhant.shete@dfki.de

[†] Presented at the 14th EASN International Conference on "Innovation in Aviation & Space towards sustainability today & tomorrow", Thessaloniki, Greece, 8–11 October 2024.

Abstract: This study presents a deep learning-based prediction system with an elevated approach to prevent tip-over incidents on planetary exploration rovers, enhancing their operational safety and reliability. Planetary rovers, critical for space exploration missions, must navigate through uneven surfaces and terrains with undefined interaction properties. Future planetary rovers must navigate harsher terrains, like steep craters and caves, to access critical scientific data, significantly risking tip-over in any state of operational control. The proposed system employs linear accelerations and angular velocities measured by the accelerometer and the gyroscope of the Inertial Measurements Unit (IMU) to monitor the rover's dynamic behavior and stability while navigating the environment. By leveraging deep learning algorithms, the system evaluates predictions and true measurements in real time to identify potential tip-overs. Additionally, the system provides the possibility to adjust the rover's motion to prevent failure. The efficacy of this prediction-based approach is validated through simulations and field tests on two robotic platforms, the Asguard v4 and Coyote 3 rovers, demonstrating its capability to reduce the incidence of tip-overs under various challenging conditions. The integration of this system aims to extend the operational lifespan of rovers, optimize mission outcomes, and enhance the overall safety of planetary exploration missions.

Keywords: tip-over prediction; rover instability; deep learning; sequence modeling; rover safety; inertial measurement units (IMUs); terrain traversal

check for updates

Academic Editors: Spiros Pantelakis, Andreas Strohmayer and Nikolaos Michailidis

Published: 14 March 2025

Citation: Shete, S.; Domínguez, R.; Selvaraju, R.; Álvarez, M.D.L.; Kirchner, F. Prediction-Based Tip-Over Prevention for Planetary Exploration Rovers. *Eng. Proc.* **2025**, *90*, 44. https://doi.org/10.3390/engproc2025090044

1. Introduction

Stability during navigation is a critical challenge in space robotics, where rovers must traverse a variety of challenging terrains [1]. A loss of stability can lead to tip-over events, threatening mission success or causing irreversible damage to equipment. Communication delays in extraterrestrial missions challenge a rover's capability to predict and prevent such incidents in real time.

Planetary rovers are generally dominated by wheeled systems [2,3] offering a great balance between energy consumption and distance traveled while achieving simplicity and robustness. Compared to other locomotion modalities, such as legged or hopping devices, wheels are much more limited in the size of obstacles they can traverse. Legged systems have advanced greatly in the last decade, achieving robust mobility in a variety of terrains.

Nevertheless, these platforms remain more complex than wheeled ones and include a larger number of potential critical failure points for their basic mobility. Moreover, their efficiency in balancing energy consumption with distance traveled falls significantly short compared to wheeled systems. Considering simplicity, robustness, obstacle negotiation, and energy efficiency, designers have explored hybrid locomotion modes combining wheels and legs, such as rimless wheels, also known as *whegs*. This simple device offers a good balance between the simplicity and energy efficiency of the wheel and the capacity of negotiating obstacles with legs.

In our work, we identified that with little training, an operator can efficiently negotiate rocky, steep terrain with such systems. Intuitively, the operator is able to identify strategies to achieve efficient foot placement to climb obstacles and the risk in certain situations, like an incipient tip-over or the possibility of a foot getting trapped. Nevertheless, such skills, easily learned by a human operator after some hours of training, are not part of the Guidance Navigation and Control (GNC) of such systems.

Problem Definition

The challenges outlined in the work by Dettmann et al. [4] regarding rover stability on steep slopes, particularly the reliance on the manipulator's continuous arm adjustment for center of mass balance to prevent tipping over, raise critical concerns about system robustness. Predicting tip-over in such cases becomes an important issue. This poses a potential threat to rovers, risking severe system damage when tipped over.

This paper addresses two key challenges: The first challenge is evaluating the reliability of deep neural networks (DNNs) in the real-time prediction of unseen events by detecting subtle, significant changes in IMU data. The second challenge focuses on the system's ability to effectively utilize accurately predicted events in real time and prevent rover tip-overs by leveraging a selective set of output features and issuing timely motion commands.

This publication presents the results of a predictive approach aimed at detecting the starting moment of tip-overs on rimless rovers early enough to abort the next failure-causing motion commands. The approach uses, at its core, a machine learning model trained to predict the IMU data flow of the system in a nominal state. With sufficient accuracy, the forward model can closely replicate the measurements provided by the IMU. The shorter the prediction horizon of the model, the easier the problem of yielding accurate predictions. Thus, the strategy to yield better results is to compare the immediate predictions and detect errors in these. Large errors in the predictions of an accurate model trained only on nominal data are indicators of an upcoming unexpected, potentially hazardous event: in this case, a tip-over.

2. Related Work

This section describes the methodology established by De Lucas Alvarez [5], which is used in this work. The method proposed utilizes an autoencoder alongside threshold-based anomaly detection to detect incipient tip-over events on the Asguard v4 [6] rover navigating along an artificial lunar analog slope. Rauch et al. [7] applied sensor feedback prediction to monitor robotic behavior using IMU data to detect discrepancies and trigger emergency actions, while Köhler et al. [8] emphasized real-time sensor predictions for adapting robot behavior in unpredictable terrains, aligning with the tip-over prevention system developed in this research. The system utilized by De Lucas Alvarez employs a Long Short-Term Memory (LSTM)-based encoder–decoder framework [9] and a recurrent neural network (RNN) model, effective in sequence prediction tasks, particularly for time-series sensor data during simulated lunar terrain traversals. The LSTM architecture captures temporal dependencies in sensor data, enabling the model to predict potential hazards like tip-overs and facilitate proactive risk mitigation. Additionally, the use of Mean Squared Error (MSE) and Mean Absolute Error (MAE) detectors [10] enhances hazard detection.

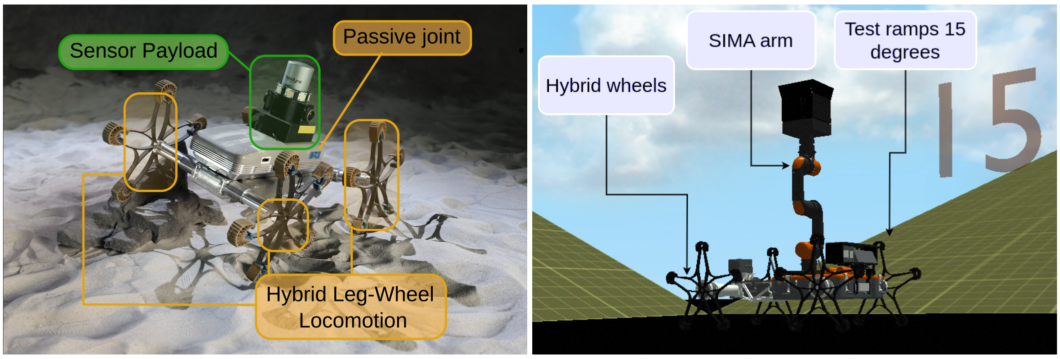
Contributions: This study evaluates five advanced models—Bi-LSTM, BiLSTM-Att, Conv-BiLSTM, GRU, and GRU-Att—against LSTM for real-time prediction at a 100 Hz sampling rate, synchronized with IMU data acquisition to enable timely hazard detection. The primary contribution of this work lies not only in algorithmic development but also in onboard integration and testing, demonstrating its real-world application. The approach includes a proactive stop mechanism to avert tip-overs utilizing selective features to optimize accuracy and efficiency, making it well suited for resource-constrained environments like lunar exploration.

3. Methodology

3.1. Sensor Information

The Asguard v4 rover is a wheel–leg hybrid rover that can be used to navigate unstructured environments, including harsh ones like lava caves, for multiple hours without an external power supply. Its hybrid wheel–legs are conceived to facilitate the locomotion over obstacles in an unstructured environment but have also proved to be effective for climbing stairs. The rover features an additional passive joint connecting the back wheel axis and the main body; the passive joint helps the system maintain contact with the ground while traversing obstacles.

The rover’s base includes encoders in each of the wheels and an Xsens Mti-28A53G35 IMU, which allows the system to realize rough self-localization through wheel-based odometry. In addition, a sensor module is attached to the system, which is used for remote operations and mapping. The sensor module is composed of a Velodyne 32 LiDAR, a stereo camera with 2 Entaniya Fisheye 220 lenses, and LEDs for illumination in dark areas such as caves. A visual overview of the rover and its equipment is provided in Figure 1a. In addition to these sensors, a satellite navigation receiver is often mounted on top of the sensor module. Figure 1b shows the second rover, Coyote 3, which has the same locomotion approach as Asguard v4 and has an arm attached. Coyote 3 was used to test the performance of the GRU model in simulation.



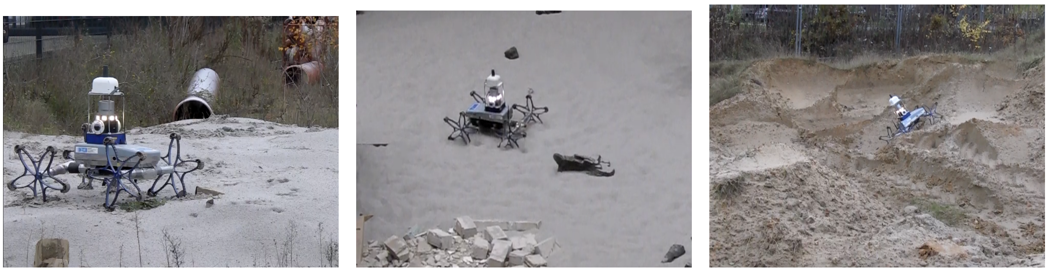
(**a**) The test rover Asguard v4 (**b**) Coyote 3 rover with SIMA arm in simulation

Figure 1. The platforms used to test the algorithm.

3.2. Dataset

In this research, we utilize IMU data to tackle the potential problem. In the domain of rover and vehicle stability, an IMU dataset embodies a wealth of information crucial for tip-over detection. By capturing measurements of linear acceleration and angular velocity across multiple axes, the IMU offers real-time insights into the dynamic behavior of vehicles navigating uneven terrain or encountering slopes [11]. Integrated with control systems, IMU data facilitate proactive adjustments to the vehicle's configuration or trajectory, pre-empting potential tipping incidents [12]. Moreover, IMU datasets serve as foundational inputs for feature extraction and analysis, enabling the characterization of motion patterns and dynamics essential for stability control algorithms [13]. Leveraging machine learning techniques, historical IMU data empower the training of predictive models capable of identifying precursors to tipping events, thereby bolstering safety and stability during exploration operations in challenging environments [14].

Experiments were conducted at the German Research Center for Artificial Intelligence GmbH (DFKI) over four trials [15], with a total runtime of approximately 40 min. Note that all datasets are publicly available. Each trial generated datasets comprising both tip-over and non-tip-over events, providing a robust basis for training and evaluating neural networks (NNs) for tip-over detection. The trials were conducted in varied environments, including indoor and outdoor settings of natural and artificial sandy terrain, exposing the model to different compactness characteristics. This variability enhanced the model's capability to learn fine-grained terrain characteristics, improving its adaptability and real-world performance. The dataset, as shown in Figure 2, consisted of approximately 161,000 samples collected from precisely calibrated rover sensors. The vital parameters included timestamps, triaxial accelerometer and gyroscope readings, and their magnetic field. The focus on triaxial gyroscope features is critical for improving the accuracy and granularity of the tip-over detection process. The test dataset incorporated multiple tip-over instances, further refining the model's predictive accuracy and robustness in detecting such events. Additionally, the training datasets were collected in a simulated environment with ramps inclined at 10, 15, and 25 degrees with the rover and manipulator maneuvered to simulate various non-tip-over scenarios, as depicted in Figure 1b.



(**a**) Soft, compact sand (**b**) Soft, loose sand (**c**) Hard, compact sand

Figure 2. Images exhibiting the data collection experiments conducted both indoors and outdoors at DFKI premises.

3.3. Neural Networks

This section describes how NNs were used in the experiment for learning sensitive IMU data and generating predictions:

- Bidirectional Long Short-Term Memory (Bi-LSTM) and Convolutional Bidirectional Long Short-Term Memory (Conv-Bi-LSTM): The Bi-LSTM architecture uses stacked LSTM cells to process sequential data in both forward and backward directions, capturing temporal dependencies from past and future contexts [16,17]. While training a Bi-LSTM is computationally expensive, it enhances the ability to learn intricate patterns in sequential data. The Conv-BI-LSTM model combines convolutional layers with an LSTM encoder–decoder network [16,18]. The convolutional layers extract local spatial features, which are then processed by the Bi-LSTM to capture long-term dependencies.
- Gated Recurrent Unit (GRU): GRU is a recurrent neural network architecture similar to LSTM, designed for sequential data modeling [19]. GRU uses two gates—an update

gate and reset gate—in place of LSTM's three gates. It has a simpler architecture and offers computational efficiency, making it particularly effective for smaller datasets where long-term dependencies are less critical [20].

- Bi-LSTM and GRU with Attention Mechanism: Integrating an attention mechanism into sequence-to-sequence models improves the network's ability to focus on irregular patterns in input data [21]. Applied to both Bi-LSTM and GRU encoder–decoder networks, this mechanism enables dynamic adaptation by selectively focusing on relevant parts of the input data at each time step [22].

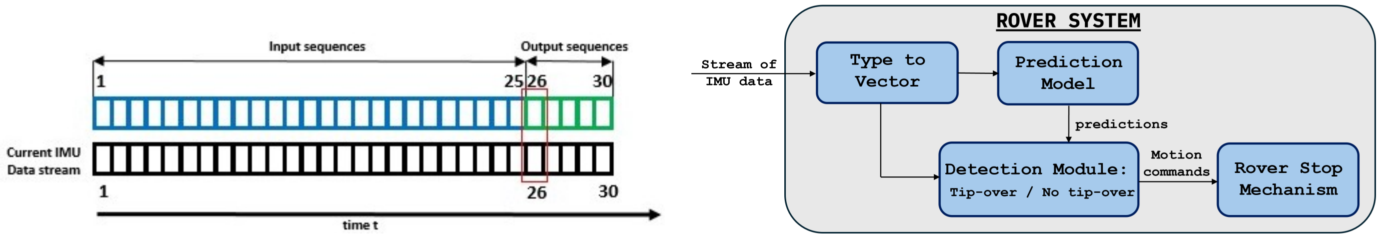
3.4. Evaluation Metrics

When analyzing IMU data for tip-over detection, it is critical to incorporate both gyroscopic and accelerometer values. Gyro[2] measurements capture rotational velocity around the *z*-axis, crucial for detecting tip-over events, while accelerometer readings reveal movement patterns across various terrains. This combination enables the prediction model to understand complex dynamics and interactions, enhancing environmental comprehension. Tipping along the *x*- or *y*-axis often results in changes in angular velocity around the *z*-axis [23], making gyro[2] readings meaningful for assessing rover stability and thereby reducing the number of evaluation features. MAE is a suitable metric for evaluating model performance, particularly in safety-critical applications. It provides a linear measure of error, treating all deviations equally, ensuring consistent and interpretable evaluations. In this context, MAE was calculated between the true and predicted gyro[2] readings, capturing subtle rotational deviations vital for tip-over detection. This approach allows for precise and balanced error minimization, enhancing the model's reliability across diverse scenarios.

3.5. Model Architecture

Our focus on sequential models required the dataset to be preprocessed according to the problem domain. To optimize training, the dataset was aligned with the structural requirements and operational dynamics of sequential models. By doing so, we improved the effectiveness and performance of our model training process.

We adhered to a structured approach similar to that in Alvarez [5], where each input sequence consists of 25 samples with a time step of 1 and 5 output sequences, as detailed in Figure 3a. This setup aligns target sequences with the following 5 output sequences, enabling the detection of tip-over occurrences at every time step and enhancing the model's robustness, critical for our experimental framework. To optimize the model for real-time application, the prediction model was quantized to accelerate inference times. This enabled the synchronization of input IMU data at 100 Hz, ensuring that the rover could process sensor data and make timely predictions for tip-over detection without delay.



(**a**) (**b**)

Figure 3. Data processing terminology and model architecture. (**a**) IMU data sequences (blue: input; green: output, red: error calculation) and comparison between two IMU samples during postprocessing. (**b**) Tip-over prevention model architecture.

The Asguard v4 framework addresses multiple challenges, including anomalous motion detection and sensor error compensation. The rover operates on the ROCK (Robot

Construction Kit) [24], an open source software framework based on the Orocos RTT (Real-Time Toolkit), providing the necessary tools for setting up and operating high-performance, robust robotic systems for diverse applications in research and industry. Figure 3b illustrates this functionality. During exploratory activities, the rover generates a constant stream of data, which its onboard computer converts into a structured format via the Type-to-Vector library, making it interpretable and usable. Data are transmitted using ports in the ROCK framework, facilitating seamless communication between components. The transformed IMU matrices are transmitted across the communication channels to the tip-over prediction module for real-time evaluation. Preprocessing, as outlined in Figure 3a, optimizes the data for efficient utilization within the NN model, enhancing predictive performance. The selection of 25 input sequences to predict the subsequent 5 balances temporal dependencies and computational efficiency, enabling short-term trend anticipation while minimizing error accumulation. To address the complexities of hyperparameter tuning, we employed HyperOpt, an automated machine learning (AutoML) tool, for model training and refinement [25]. By defining a comprehensive search space encompassing batch size, dropout rate, learning rate, and architectural parameters like GRU units, LSTM units, convolutional kernel size, and layers, we efficiently explored various hyperparameter combinations [26].

The predicted sequences were transmitted to the output ports for postprocessing, where they were compared with current IMU samples, as shown in Figure 3a, to calculate errors. Since the training data lacked tip-over events, the predicted values established a decision boundary. Outliers in the test dataset were identified through density plots of MAE distributions, with model-specific thresholds that vary across models. Values exceeding these thresholds were classified as "tip-over" events. Upon detection, motion commands are sent to the rover's motion controller to stop, preventing potential tip-overs and mitigating system damage. This proactive approach enables real-time preventive actions, ensuring the rover's stability and system integrity.

4. Results

This section presents findings from the experimental evaluation of the tip-over detection system, focusing on its ability to identify potential tip-over occurrences in dynamic environments. Our analysis focuses on key performance metrics and graphical representations, shedding light on the effectiveness and reliability of the detection system under varying conditions. The analysis of model performance highlights the significance of different evaluation metrics in assessing detection efficacy. Figure 4 presents a comparative analysis of prediction performance across different NN architectures for various hyperparameter settings on a validation dataset.

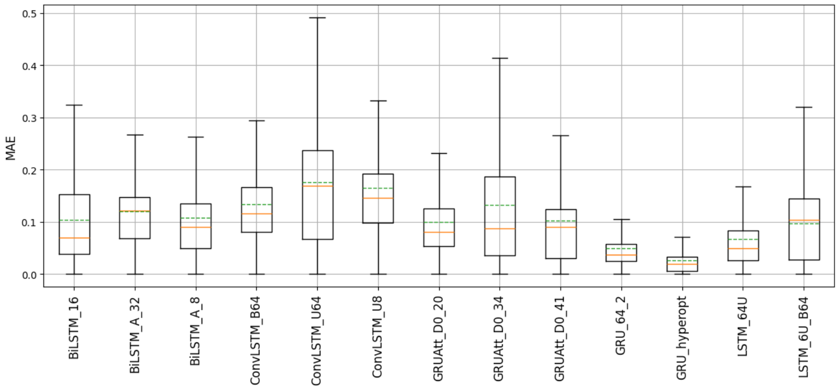


Figure 4. Comparison of models' prediction performances on validation dataset (orange: median, green: mean).

Figure 5 offers a detailed evaluation of the model's performance in predicting tip-over events using gyro[2] data. The top graph illustrates the comparison between the true values (actual gyro[2] measurements) and the predicted values generated by the model. In addition, the MAE between the true and predicted values highlights the model's prediction accuracy, with spikes in error occurring during periods of significant deviation, typically around potential tip-over events. When the MAE surpasses the threshold, the system flags a tip-over, marked by green dots. The bottom graph shows the ground truth labels as a binary classification, providing a reference for validating the model's prediction performance.

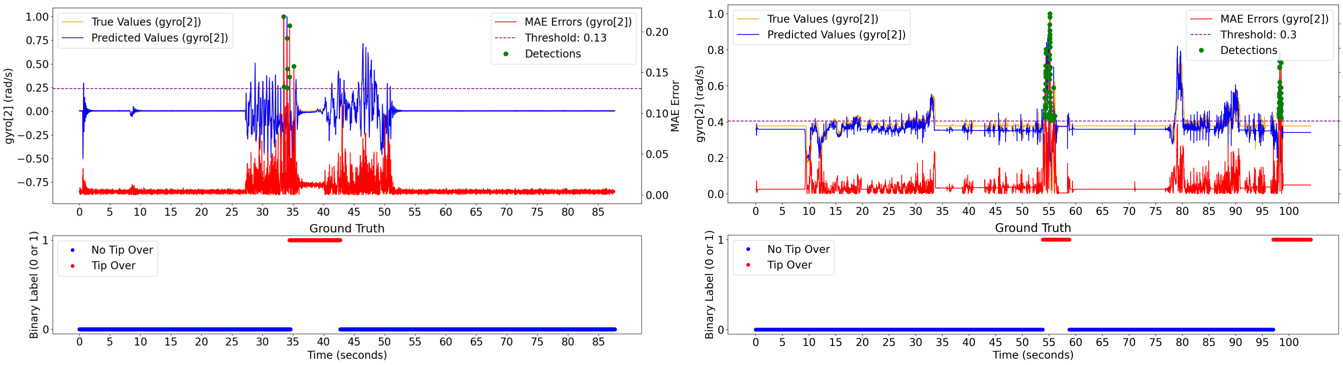


Figure 5. Comparison of GRU model's prediction performance in Asguard v4 and Coyote3 (simulation) systems with ground truth for gyroscope angular velocity (gyro[2]).

Aligning the two graphs enables a direct comparison between detected tip-over events (green dots) and true occurrences (red sections), highlighting the model's accuracy and reliability in forecasting the tip-over events. This predictive capability enables the system to preemptively send motion commands to the rover, allowing the rover to stop before a tip-over occurs, thereby preventing potential damage to the system.

In addition to the graphs, Table 1 compares various neural network models for tip-over prediction across two test datasets. The "true" values represent the actual tip-over event times, while the "predicted" values indicate the model's prediction times, highlighting how early the predictions were relative to the actual events and the count of correctly predicted occurrences. GRU_hyperopt predicts a tip-over at 33.49 s in Dataset 1, preceding the actual event at 34.40 s, with one event correctly detected. Similarly, in Dataset 2, it predicts an event at 23.64 s against an actual 24.90 s. These results demonstrate the efficacy of hyperparameter optimization in enhancing prediction accuracy. Conv-Bi-LSTM_B128_U8 and Conv-Bi-LSTM_B64_U64 perform well, particularly in Dataset 1, where it closely matches the actual tip-over time. However, in Dataset 2, Conv-Bi-LSTM_B128_U8 slightly overestimates the timing (25.37 s predicted vs. 24.90 s actual). The model's sensitivity to sudden jerks or oscillations, especially on harder surfaces, can lead to false positives. For instance, Bi-LSTM_B16_U64 and GRU-Att_B128_U16 overestimate the number of events in Dataset 2, predicting two events instead of one.

The rover's performance varies significantly with surface type. On sand, the movement is smoother due to the damping effect of the soft surface, resulting in fewer abrupt changes in acceleration or tilt. On harder surfaces, such as rocky or compact terrain, sinusoidal bumpiness and sudden jerks are more pronounced, leading to potential misinterpretation of terrain-induced bumps as tip-over events. These discrepancies can be mitigated by dynamically adjusting the detection threshold based on surface type. Increasing the threshold for harder surfaces could reduce false positives while maintaining early detection capabilities. GRU-hyperOpt proved the most reliable model, ensuring timely and accurate tip-over detection, critical for preventive actions in safety-critical rover operations.

Table 1. Comparison of true and predicted tip-over events across two datasets (bold text indicates the best-performing model among all tested models).

Model	Batch Size	Units	Test Dataset 1		Test Dataset 2	
			True (sec) /No. of Events	Predicted (sec) /No. of Events	True (sec) /No. of Events	Predicted (sec) /No. of Events
Bi-LSTM	16	64	34.40 (1)	33.48 (2)	24.90 (1)	22.89 (2)
Bi-LSTM-Att	8	8	34.40 (1)	29.00 (2)	24.90 (1)	25.50 (1)
Conv-Bi-LSTM	64	64	34.40 (1)	33.53 (1)	24.90 (1)	25.58 (1)
Conv-Bi-LSTM	128	64	34.40 (1)	34.40 (1)	24.90 (1)	25.64 (2)
Conv-Bi-LSTM	128	8	34.40 (1)	34.01 (1)	24.90 (1)	25.37 (1)
GRU-hyperopt	**8**	**64**	**34.40 (1)**	**33.49 (1)**	**24.90 (1)**	**23.64 (1)**
GRU-Att	128	16	34.40 (1)	34.98 (1)	24.90 (1)	25.57 (2)
LSTM	64	6	34.40 (1)	35.41 (1)	24.90 (1)	25.80 (1)

Abbreviations: Conv = convolutional, Bi = bidirectional, Att = attention.

To further validate the tip-over prediction model, testing was conducted in a simulated environment using a more complex rover system, Coyote 3, equipped with a manipulator. This setup increased the system complexity, challenging the prediction model. Testing on Coyote 3 demonstrated the model's platform independence and generalization capabilities, ensuring robustness across different rover configurations. The same prediction model used in previous tests was employed for this evaluation, and its performance is shown in Figure 5(right).

5. Conclusions

In conclusion, this study presents a robust framework for real-time tip-over detection in autonomous rovers, quantizing a model for synchronized IMU data at 100 Hz to enable real-time prediction and timely preventive actions. The GRU-hyperopt model demonstrated reliability in predicting and detecting tip-overs across different datasets, proving its effectiveness in safety-critical scenarios. GRU is favored over LSTM due to its more streamlined architecture, which minimizes computational complexity, memory usage, and training time. This efficiency is crucial for real-time applications, enabling quick decision making and resource optimization, thereby outperforming state-of-the-art models. By aligning the prediction frequency with the data acquisition rate, the method ensures seamless integration with the rover's control systems, achieving a closed-loop system for tip-over prevention. The model was further validated in a simulated environment using the Coyote 3 mobile manipulation system, which includes a manipulator for increased system complexity. This testing demonstrated the model's ability to generalize across different platforms and scenarios, ensuring robustness and adaptability to varying operational conditions. The models still need to be thoroughly tested across a wide range of variational test scenarios.

Future research could integrate surface-type classification for dynamic threshold adjustments in real time based on data like wheel slip, vibration frequency, or terrain roughness, enhancing the model's robustness and reducing false alarms. Further testing in diverse and extreme real-world scenarios, including planned trials in an analogue planetary surface environment, will validate the system's scalability and robustness. These advancements will not only ensure safer autonomous operations but also expand the applicability of tip-over prediction systems across various robotics domains.

Author Contributions: Conceptualization, methodology, approach development, data collection, robotic framework integration and deployment, testing, analysis, and writing—original draft, S.S.; supervision, robotic framework integration and deployment, and data collection and testing, R.D.; simulation, R.S.; supervision, M.D.L.Á.; supervision, F.K. All authors have read and agreed to the published version of the manuscript.

Funding: The PerSim project is funded by the Space Agency of the German Aerospace Center with federal funds of the German Federal Ministry of Economic Affairs and Climate Action (BMWK) in accordance with the parliamentary resolution of the German Parliament, grant no. 50RA2124.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original data presented in the study are openly available in Zenodo at https://doi.org/10.5281/zenodo.14198120 (DOI: 10.5281/zenodo.14198120), accessed on 12 March 2025.

Conflicts of Interest: Author Siddhant Shete, Raúl Domínguez, Ravisankar Selvaraju, Mariela De Lucas Álvarez and Frank Kirchner were employed by the company Robotics Innovation Center of German Research Center for Artificial Intelligence GmbH. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Gao, Y.; Chien, S. Review on space robotics: Toward top-level science through space exploration. *Sci. Robot.* **2017**, *2*, eaan5074.
2. Farley, K.A.; Williford, K.H.; Stack, K.M.; Bhartia, R.; Chen, A.; de la Torre, M.; Hand, K.; Goreva, Y.; Herd, C.D.K.; Hueso, R.; et al. Mars 2020 Mission Overview. *Space Sci. Rev.* **2020**, *216*, 142. [CrossRef]
3. Vago, J.L.; Westall, F.; Coates, A.J.; Jaumann, R.; Korablev, O.; Ciarletti, V.; Mitrofanov, I.; Josset, J.L.; De Sanctis, M.C.; Bibring, J.P.; et al. Habitability on Early Mars and the Search for Biosignatures with the ExoMars Rover. *Astrobiology* **2017**, *17*, 471–510. [CrossRef] [PubMed]
4. Dettmann, A.; Römmermann, M.; Cordes, F. Evolutionary development of an optimized manipulator arm morphology for manipulation and rover locomotion. In Proceedings of the 2011 IEEE International Conference on Robotics and Biomimetics, ROBIO 2011, Karon Beach, Thailand, 7–11 December 2011; pp. 2567–2573. [CrossRef]
5. Alvarez, M.D.L. Safety Enhancement in Planetary Rovers: Early Detection of Tip-over Risks Using Autoencoders. *arXiv* **2024**, arXiv:2408.05602. [CrossRef]
6. Domínguez, R.; Arnold, S.; Hertzberg, C.; Böckmann, A. Internal Simulation for Autonomous Robot Exploration of Lava Tubes. In Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics, ICINCO 2018—Volume 2, Porto, Portugal, 29–31 July 2018; Madani, K., Gusikhin, O., Eds.; SciTePress: Setúbal, Portugal , 2018; pp. 154–165. [CrossRef]
7. Rauch, C.; Berghöfer, E.; Köhler, T.; Kirchner, F. Comparison of Sensor-Feedback Prediction Methods for Robust Behavior Execution. In *Proceedings of the KI 2013: Advances in Artificial Intelligence*; Timm, I.J., Thimm, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 200–211.
8. Tiedemann, T.; Berghöfer, E.; Rauch, C.; Kirchner, F. Sensor Fault Detection and Compensation in Lunar/Planetary Robot Missions Using Time-Series Prediction Based on Machine Learning. *Acta Futura* **2014**, 9–20.
9. Kao, I.F.; Zhou, Y.; Chang, L.C.; Chang, F.J. Exploring a Long Short-Term Memory based Encoder-Decoder framework for multi-step-ahead flood forecasting. *J. Hydrol.* **2020**, *583*, 124631. [CrossRef]
10. Hodson, T.O. Root-mean-square error (RMSE) or mean absolute error (MAE): When to use them or not. *Geosci. Model Dev.* **2022**, *15*, 5481–5487. [CrossRef]
11. Islam, F.; Nabi, M.M.; Ball, J.E. Off-Road Detection Analysis for Autonomous Ground Vehicles: A Review. *Sensors* **2022**, *22*, 8463. [CrossRef] [PubMed]
12. Roan, P.; Burmeister, A.; Rahimi, A.; Holz, K.; Hooper, D. Real-World Validation of Three Tipover Algorithms for Mobile Robots. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 3–7 May 2010; pp. 4431–4436. [CrossRef]
13. Yıldırım, Ş.; Arslan, E. ODE (Open Dynamics Engine) based stability control algorithm for six legged robot. *Measurement* **2018**, *124*, 367–377. [CrossRef]
14. Toledo Fuentes, A.; Kempf, F.; Kipfmüller, M.; Bergmann, T.; Prieto, M.J. Tip-Over Detection and Avoidance Algorithms as Stabilization Strategy for Small-Footprint and Lightweight Mobile Manipulators. *J. Machines* **2023**, *11*, 44. [CrossRef]

15. Shete, S.; Domínguez, R. *IMU-Based Tip-Over Dataset for Space Exploration Rover Dynamics and Stability Analysis*; Zenodo: Bremen, Germany; Genève, Switzerland, 2024. [CrossRef]
16. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to Sequence Learning with Neural Networks. *arXiv* **2014**, arXiv:1409.3215. [CrossRef]
17. Raihan, A.S.; Ahmed, I. A Bi-LSTM Autoencoder Framework for Anomaly Detection—A Case Study of a Wind Power Dataset. *arXiv* **2023**, arXiv:2303.09703.
18. Gehring, J.; Auli, M.; Grangier, D.; Yarats, D.; Dauphin, Y.N. Convolutional Sequence to Sequence Learning. *arXiv* **2017**, arXiv:1705.03122.
19. Dey, R.; Salem, F.M. Gate-Variants of Gated Recurrent Unit (GRU) Neural Networks. *arXiv* **2017**, arXiv:1701.05923.
20. Yang, S.; Yu, X.; Zhou, Y. LSTM and GRU Neural Network Performance Comparison Study: Taking Yelp Review Dataset as an Example. In Proceedings of the 2020 International Workshop on Electronic Communication and Artificial Intelligence (IWECAI), Shanghai, China, 12–14 June 2020. [CrossRef]
21. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2023**, arXiv:1706.03762.
22. Azad, M.I.; Rajabi, R.; Estebsari, A. Sequence-to-Sequence Model with Transformer-based Attention Mechanism and Temporal Pooling for Non-Intrusive Load Monitoring. *arXiv* **2023**, arXiv:2306.05012.
23. Lyu, S.; Zhang, W.; Yao, C.; Zhu, Z.; Jia, Z. Modeling and Analysis of a Reconfigurable Rover for Improved Traversing over Soft Sloped Terrains. *Biomimetics* **2023**, *8*, 131. [CrossRef] [PubMed]
24. ROCK, the Robot Construction Kit. Available online: http://www.rock-robotics.org (accessed on 12 March 2025).
25. Yang, Z.; Zhang, A. Hyperparameter optimization via sequential uniform designs. *J. Mach. Learn. Res.* **2021**, *22*, 1–47.
26. Chen, Y.; Song, X.; Lee, C.; Wang, Z.; Zhang, R.; Dohan, D.; Kawakami, K.; Kochanski, G.; Doucet, A.; Ranzato, M.A.; et al. Towards Learning Universal Hyperparameter Optimizers with Transformers. In *Proceedings of the Advances in Neural Information Processing Systems*; Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A., Eds.; Curran Associates, Inc.: New York, NY, USA, 2022; Volume 35, pp. 32053–32068.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.