Terrain Identification using Neural Networks

Submitted in partial fulfillment of the requirements for the degree of

M.Tech Integrated Computer Science and Engineering (In Collaboration with Virtusa)

by

Niketha Sabesan - 19MIC0035
Shyam R - 19MIC0017
Purushothaman S - 19MIC0057
Nivethitha - 19MIC0030

Under the guidance of

Prof. / Dr.Jagalingam P

SCOPE

VIT, Vellore.

VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

April , 2023

DECLARATION

I hereby declare that the thesis entitled “Terrain Identification using Neutral Networks" submitted by me, for the award of the mini-project in M.Tech. (Integrated) Computer Science and Engineering to VIT is a record of bonafide work carried out by me under the supervision of Prof. Jagalingam.

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place : Vellore

Date : 10.04.2023

Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled "Terrain Identification using Neutral Networks" submitted by **Niketha Sabesan(19MIC0035), Shyam R (19MIC0017) , Purusothaman S (19MIC0057), Nivethitha S(19MIC0030) from SCOPE**, VIT, for the award of the Mini- project of M.Tech. (Integrated) Computer Science and Engineering, is a record of bonafide work carried out by him / her under my supervision during period, 10. 12. 2022 to 10.04.2023, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

Place : Vellore

Date : 10.04.2023

Signature of the Guide

Internal Examiner

External Examiner

Head of the Department

Programme

ACKNOWLEDGEMENTS

We would like to take this opportunity to express my gratitude to everyone who supported me throughout the process of writing this report.

Firstly, I would like to thank my professor, Dr.Vijesh Joe C, for providing me with valuable feedback and guidance throughout the project. Your insights and expertise were instrumental in shaping the direction of this report.

I would also like to express my appreciation to the staff at the college library for their assistance in locating and accessing relevant resources for my research.

Furthermore, I extend my thanks to my classmates for their helpful discussions and contributions that greatly enhanced the quality of this report.

Lastly, I would like to express my heartfelt appreciation to my family and friends for their unwavering support, encouragement, and patience during this time. Your belief in me has been a constant source of motivation.

Thank you all for your support and encouragement throughout this process.

Student Name

Niketha Sabesan
Shyam R
Purushothaman S
Nivethitha

Executive Summary

This study sought to explore the feasibility of utilizing inertial data from human walking to identify various types of subterranean terrain. The main objective is to develop a system that gathers inertial measurement unit data from sensors attached to the lower limb to identify terrain types without visual input. To achieve this, the researchers proposed a combination of LSTM, MLP, and CNN models to analyze time series data and differentiate between various types of terrain. The methodology involves identifying points of motion or directional changes, referred to as "inertia," to demarcate different terrain types. This approach has broad implications in numerous fields for understanding the behavior of complex systems over time.

The study's potential impact is significant, particularly in the fields of robotics and prosthetics, where the ability to identify and navigate different types of terrain is crucial. Additionally, it has the potential to enhance human mobility, especially in challenging terrain , such as underground caves or rough terrain. However, there are challenges to overcome, such as the accuracy of the inertial measurement unit data and the performance of the proposed models. Overall, the study provides valuable insights into the potential of inertial data for terrain identification and highlights the need for further research to develop accurate and reliable systems for identifying different types of terrain.

CONTENTS

		Page No.
	Acknowledgement	i
	Executive Summary	ii
	Table of Contents	Iii
	List of Figures	ix
	List of Tables	xiv
	Abbreviations	xvi
	Symbols and Notations	xix
1	**INTRODUCTION**	1
	1.1 Theoretical Background	1
	1.2 Motivation	2
	1.3 Aim of the Proposed Work	3
	1.4 Objective(s) of the Proposed Work	4
Literature Survey		**5**
	2.1. Survey of the Existing Models/Work	5
	2.2. Summary/Gaps identified in the Survey	9
Overview of the Proposed System		**10**
	2.3. Introduction and Related Concepts	10
	2.4. Framework, Architecture or Module for the Proposed System(with explanation)	12
	2.5. Proposed System Model(ER Diagram/UML Diagram/Mathematical Modeling)	15
Proposed System Analysis and Design		**18**
	2.6. Introduction	18
	2.7. Requirement Analysis	19
	Functional Requirements	
	Product Perspective	
	Product features	
	User characteristics	
	Assumption & Dependencies	
	Domain Requirements	
	User Requirements	
	Non Functional Requirements	25
	Product Requirements	

2.7.1.1.1. Efficiency (in terms of Time and Space)

2.7.1.1.2. Reliability

2.7.1.1.3. Portability

2.7.1.1.4. Usability

Organizational Requirements 30

2.7.1.1.5. Implementation Requirements (in terms of deployment)

2.7.1.1.6. Engineering Standard Requirements

Operational Requirements (Explain the applicability for your work w.r.to the following operational requirement(s))

- Economic
- Environmental
- Social
- Political
- Ethical
- Health and Safety

System Requirements 40

H/W Requirements(details about Application Specific Hardware)

S/W Requirements(details about Application Specific Software)

Results and Discussion 43

References 50

APPENDIX A

List of Abbreviations

3GPP	Third Generation Partnership Project
2G	Second Generation
3G	Third Generation
4G	Fourth Generation
AWGN	Additive White Gaussian Noise

Symbols and Notations

δf	CFO
ε	NCFO

1. INTRODUCTION

1.1 Theoretical Background

This study set out to determine whether the inertial data obtained from a typical human walk might be utilized to identify different types of subterranean terrain. We want to create a system for identifying terrain that gathers inertial measurement units from sensors attached to a person's lower limb.
Nonetheless, we will test the viability of terrain identification without visual information in this project. Based on time series data, we suggest an LSTM,MLP,CNN model to distinguish between the various types of terrain

1.2 Motivation

Inertia motivation is a useful approach for terrain identification with time series data. By analyzing the fluctuations in the time series data, we can identify the points of change that indicate different types of terrain. This approach can be applied to a wide range of terrain identification problems in various fields, providing valuable insights into the behavior of complex systems over time.

1.3 Aim of proposed work

The challenge in this task is to identify different types of terrain based on the patterns and trends in the time series data. One approach to addressing this challenge is to use inertia motivation. This approach is based on the principle that the behavior of a system is determined by its inertia, which refers to its tendency to resist changes in motion or direction. Inertia motivation works by identifying the points in the time series data where the direction or velocity of the system changes, and then using these points to identify different types of terrain.

1.4 Objective(s) of the Proposed Work

The aim of this project was to investigate whether the inertial data gathered during a typical human walk could be used to differentiate between different types of underground terrain. Our objective is to develop a system that collects data from inertial measurement units attached to a person's lower limb for terrain identification. However, in this project, we will assess the feasibility of identifying terrain without relying on visual information. To achieve this, we propose using an LSTM model based on time series data to differentiate between various types of terrain.
The motivation for this approach lies in the effectiveness of inertia motivation for terrain identification with time series data. By scrutinizing the fluctuations in the time series data, we can pinpoint the points of change that indicate different types of terrain. This method has broad applicability to terrain identification challenges across several fields and can provide useful insights into complex system behavior over time.

2.Literature Survey

2.1) Survey of the Existing Models/Work

1. SVM performed better with fewer classes, whereas RF performed better with more classes in classification tasks.
2. A recurrent method can contextualize observations over lengthy periods for real-time human activity modeling.
3. A robust environment system can anticipate amputees' styles of mobility and estimate environmental aspects at least 0.6 seconds before actual locomotion changes.
4. A convolutional neural network is used to classify different types of terrain, achieving an accuracy greater than 98% for both real and virtual situations, outperforming state-of-the-art terrain classification methods for legged robots.
5. This paper introduces visual sensors to build a pattern recognition system for the soft lower limb exosuit robot.

2.2)Summary/Gaps identified in the Survey

The data collection was limited to only five terrains, and the database had an unbalanced male-to-female ratio. Deep learning research lacks information on how the best system parameters were determined. IMU and EMG inputs have delays, so user-independent sensors may be helpful in accurately predicting amputee locomotion. The virtual terrains were created using Gazebo simulation's surface parameters, and the damping value remained constant. The pattern recognition system aids faster and more accurate conversion between movement modes, and delayed conversion can lead to out-of-time responses.

3.Overview of the Proposed System

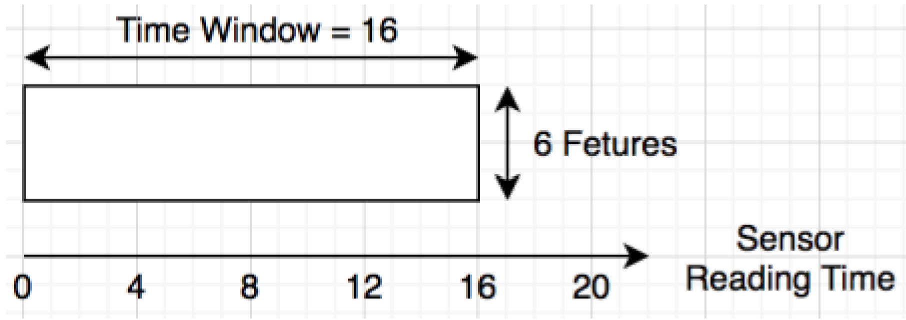
3.1) Introduction and Related Concepts

To apply inertia motivation to terrain identification with time series data, we first need to preprocess the data to remove any noise or outliers. We can then calculate the first and second derivatives of the time series data to obtain the velocity and acceleration profiles, respectively. We can then use the velocity and acceleration profiles to identify the points in the time series data where the direction or velocity of the system changes.

3.2)Framework, Architecture or Module for the Proposed System(with explanation)

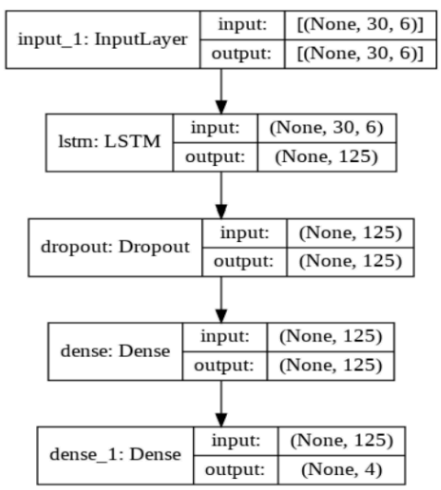
To achieve terrain identification from time-series data, the framework employs a recurrent neural network (RNN), specifically the Long Short Term Memory (LSTM) network, which can eliminate the vanishing gradient problem present in traditional neural networks. The data set is explored and visualized to gain insights and remove noise that may impair model performance. A time window is

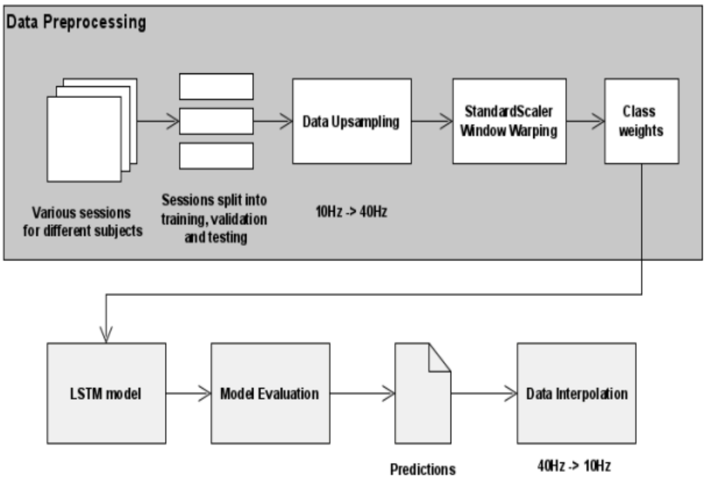
selected to construct the input for the LSTM network, with a shape of (sample size, time window, feature size = 6).
In case of highly imbalanced data sets, a customized down sampling technique is implemented to reduce the bias of the LSTM network. The framework constructs various LSTM networks and performs hyper-parameter tuning to identify the best-performing model. Numpy, pandas, sklearn, and matplotlib are used for data exploration, visualization, and preprocessing. Python random is utilized for developing customized down sampling, while Keras is utilized for building the LSTM networks.



3.3)Proposed System Model(ER Diagram/UML Diagram/Mathematical Modeling)

The Sequential model, which consists of a stack of levels with each layer having precisely one input and one output, is the first thing we build. Next, an LSTM layer with 125 neurons and an input layer (30,6) are created. To lessen the model's overfitting, a dropout layer is applied. Then, we employ one dense layer activated by ReLU and another dense layer activated by softmax. The size of the dataset is changed to the appropriate output size using the final Dense layer. Due to the fact that we have four class labels, the output size for our work will be 4. (0, 1, 2, and 3). It is shown how our model is structured.





4.Proposed System Analysis and Design

Introduction

Requirement Analysis

Functional Requirements

Product features

Data Collection: When a person is walking on various types of underground terrain, the system should be able to gather inertial measurement unit (IMU) data from sensors mounted to their lower limbs.

Data preprocessing: Before further processing, the system should clean up and normalize the IMU data it has collected.

Model Development: Based on the preprocessed IMU data, the system should create a machine learning model (LSTM, CNN, and MLP) that can categorise the various varieties of subterranean terrain.

Model Training: Using a dataset that contains IMU data gathered from various types of subsurface terrain, the system should be able to train the generated model.
Model Validation: By testing the trained model on a different dataset, the system should confirm the precision and dependability of the model.

Feature extraction: The system should extract relevant features from the preprocessed data to use as input for the machine learning models.
Classification: The system should use machine learning models to classify the different types of subterranean terrain.
Visualization: The system should be able to visualize the classification results.

User characteristics
Assumption & Dependencies

Domain Requirements

Machine Learning: Knowledge of different types of machine learning models, including Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), and Multilayer Perceptron (MLP), and their applications in time-series data analysis.

Data Science: Proficiency in data collection, cleaning, preprocessing, and visualization.

Software Development: Skills in software development and programming to create a user-friendly system that can gather, store, and process inertial data and run machine learning algorithms.

Domain-specific knowledge: Familiarity with subterranean terrain, including different types of surfaces, their characteristics, and their impact on lower-limb prosthetics.

User Requirements

Amputees: The primary users of the system will be amputees who use lower-limb robotic prosthetics. Their requirements may include:

- Comfortable and safe prosthetic use
- Accurate identification of subterranean terrain to adjust the prosthetic's behavior
- Minimal disruption to their daily activities

Non Functional Requirements

Product Requirements

Accuracy: The system should be able to accurately identify different types of terrain, including flat surfaces, slopes, hills, valleys, rocky terrain, and more, with a high degree of accuracy

Speed: The system should be able to identify terrain quickly, in real-time, so that users can make informed decisions about how to navigate through a given area.

Durability: The system should be able to withstand harsh environmental conditions, including extreme temperatures, moisture, and exposure to the elements.

Ease of Use: The system should be user-friendly, with a simple interface that allows users to quickly access and interpret terrain data.

Portability: The system should be lightweight and portable, so that users can easily take it with them on outdoor adventures.

Connectivity: The system should be able to connect to other devices, such as smartphones or GPS systems, to provide users with a comprehensive understanding of their surroundings.

Efficiency (in terms of Time and Space):

Data Processing: The system should be optimized to process large amounts of data quickly and efficiently. This can be achieved through the use of advanced algorithms, parallel processing, or other techniques that minimize processing time and maximize accuracy.

Data Storage: The system should be designed to store terrain data in a compact and efficient manner, without sacrificing accuracy or resolution. This can be achieved through the use of compression algorithms, data filtering techniques, or other methods that reduce the overall storage requirements.

Sensor Integration: The system should be able to integrate with a variety of sensors, such as cameras, lidar, or radar, to collect data on the terrain. This can improve the overall efficiency of the system, as it can reduce the amount of time required to collect data manually.

Real-Time Processing: The system should be able to process data in real-time, so that users can quickly access and interpret terrain data as they move through an area. This can improve efficiency by reducing the amount of time required to stop and analyze data.

Lightweight Design: The system should be designed to be lightweight and portable ,without sacrificing functionality or accuracy. This can improve efficiency by reducing the amount of time required to set up and use the system

Reliability:

Data Quality: The neural network should be trained on high-quality, accurate data to ensure that it can make accurate predictions. This can be achieved through the use of high-quality sensors, rigorous data filtering, and other methods that improve the quality of the input data.

Testing: The neural network should be rigorously tested in a variety of environments and conditions to ensure that it can perform reliably under different circumstances. This can help to identify and address potential issues before they become a problem in the field.

Regular Maintenance: The system should be regularly maintained, including updating the neural network with new data and retraining the network as necessary. This can help to ensure that the system remains reliable over time and that it can continue to make accurate predictions.

Redundancy: The system should be designed with redundancy in mind, such as using multiple sensors to ensure that data is being collected accurately and consistently. This can help to mitigate the risk of sensor failure or data corruption.

User Training: Users should be trained on how to use the system properly, including how to interpret data and troubleshoot potential issues. This can help to ensure that the system is being used correctly and that any issues are addressed promptly.

Portability

Compact Design: The neural network should be designed to be as compact and efficient as possible, without sacrificing accuracy or functionality. This can be achieved through the use of optimized algorithms, data compression techniques, or other method that reduce the overall size of the neural network.

Low-Power Hardware: The neural network should be designed to run on low-power hardware, such as mobile devices or embedded systems. This can improve portability by reducing the size and weight of the system, and by allowing users to operate the system for longer periods of time without needing to recharge the battery.

Cloud-Based Processing: The neural network can be designed to use cloud-based processing, where the processing power is provided by a remote server. This can improve portability by reducing the hardware requirements of the system, allowing users to access the system from a wide range of devices.

Wireless Connectivity: The system should be designed to connect wirelessly to other devices, such as smartphones or tablets, to allow users to access and control the system remotely. This can improve portability by allowing users to operate the system from a distance, or by allowing the system to be used in areas where direct access is not possible.

User-Friendly Interface: The system should be designed with a user-friendly interface that is easy to use and navigate, even for users who are not familiar with neural networks. This can improve portability by reducing the learning curve for users, and by allowing the system to be used in a wider range of applications.

Usability:

User-Friendly Interface: The system should be designed with a user-friendly interfac that is easy to use and navigate, even for users who are not familiar with neural networks. This can be achieved through the use of clear, concise labels, intuitive icons, and other design elements that make it easy for users to understand and interact with the system.

Real-Time Feedback: The system should provide real-time feedback to users, allowing them to quickly and easily understand the results of the terrain identification process. This can be achieved through the use of visualizations, graphs, or other methods that make it easy for users to interpret the data.

Customizable Settings: The system should allow users to customize the settings to meet their specific needs and preferences. This can include adjusting the sensitivity of the system, changing the input parameters, or other methods that allow users to tailor the system to their needs.

Easy Data Input: The system should make it easy for users to input data, either through direct data entry or through the use of automated data collection methods. This can help to reduce errors and improve the accuracy of the system.

Training and Support: Users should be provided with training and support to ensure that they can use the system effectively. This can include online tutorials, documentation, or other resources that help users understand how to use the system.

Organizational Requirements :

Implementation Requirements (in terms of deployment):

Data Collection: High-quality data is crucial for training and validating the neural network used in the terrain identification system. The system should be designed to collect data from reliable and accurate sources, such as high-resolution cameras, LIDAR sensors, or other relevant devices.

Data Processing: The data collected by the system must be processed and pre-processed to remove any noise or inconsistencies. This can involve filtering, feature extraction, or other data processing techniques that improve the accuracy and reliability of the data.

Neural Network Architecture: The neural network architecture must be carefully designed to suit the specific requirements of the terrain identification system. The architecture may include layers such as convolutional layers, pooling layers, and fully connected layers, among others.

Training and Validation: The neural network must be trained and validated using high-quality data that represents the conditions under which the system will be used. This process may involve several iterations of training and validation to optimize the performance of the neural network.

Hardware and Software Requirements: The hardware and software requirements for implementing a terrain identification system using neural networks may include specialized hardware such as GPUs or dedicated computing clusters, as well as software tools such as TensorFlow or PyTorch.

Integration with Existing Systems: The terrain identification system may need to be integrated with other systems, such as autonomous vehicles or drones, to provide real-time data analysis and decision-making capabilities.

Testing and Evaluation: Once the system is implemented, it must be thoroughly tested and evaluated to ensure that it meets the performance and accuracy requirements specified in the design phase.

Engineering Standard Requirements:

Data Quality and Accuracy: The data used for training and validation must be of high quality and accuracy. The system should be designed to ensure that the data is collected, processed, and validated in accordance with relevant engineering standards, such as ISO 9001.

System Performance and Reliability: The system must meet performance and reliability requirements as specified in relevant engineering standards, such as IEEE 12207. This may involve testing the system under various operating conditions to ensure that it meets the specified performance and reliability criteria.

System Safety and Security: The terrain identification system must be designed and implemented with safety and security considerations in mind. Relevant engineering standards, such as IEC 61508 or ISO 26262, may be applied to ensure that the system is safe and secure for its intended use.

Software Quality and Testing: The software used in the terrain identification system must be designed and tested in accordance with relevant engineering standards, such as ISO 25010. This may involve conducting software quality assessments, code reviews, and other testing and verification activities to ensure that the software meets the specified quality criteria.

Documentation and Traceability: The system design and implementation must be documented in accordance with relevant engineering standards, such as ISO/IEC 12207. This documentation should include detailed descriptions of the system architecture, software design, testing procedures, and other relevant information.

Operational Requirements :

1. **Economic:**

   Agriculture: Terrain identification using neural networks can help farmers optimize their crop yields by identifying the best areas of their fields for planting different crops. This can reduce costs by minimizing the use of fertilizers, pesticides, and other inputs, while maximizing yields.

   Construction: Terrain identification using neural networks can help construction companies identify the best locations for building foundations, roads, and other infrastructure. This can reduce costs by minimizing the need for expensive site preparation, and by improving the stability and longevity of the built structures.

   Mining: Terrain identification using neural networks can help mining companies identify the most promising locations for mineral deposits. This can reduce exploration costs by minimizing the need for costly and time-consuming drilling and sampling activities.

   Transportation: Terrain identification using neural networks can help transportation companies optimize their routes and schedules, based on the topography and terrain of the areas they operate in. This can reduce fuel consumption, travel time, and transportation costs, while improving safety and reliability.

   Environmental Conservation: Terrain identification using neural networks can help conservationists identify areas that are at risk of environmental degradation, such as erosion or deforestation. This can help guide conservation efforts and prevent the economic and social costs associated with environmental degradation.

2. **Environmental:**

   Habitat Mapping: Terrain identification using neural networks can help map out habitats for endangered species or those that require conservation efforts. By identifying the terrain features of these habitats, conservationists can design strategies that will help preserve these areas

Land-use Planning: Terrain identification using neural networks can help in identifying areas that are most vulnerable to environmental degradation or areas that can be designated as conservation areas. This information can be used in designing land-use plans that balance economic development with environmental conservation.

Disaster Response: Terrain identification using neural networks can help in predicting and responding to natural disasters such as floods, landslides, and wildfires. By identifying terrain features such as slopes, vegetation cover, and water bodies, disaster response teams can better understand the potential impacts of these events and respond appropriately.

Climate Change: Terrain identification using neural networks can help in modeling the effects of climate change on ecosystems and their response to these changes. By identifying areas that are most susceptible to changes in temperature, precipitation, and other climate variables, scientists can predict how ecosystems will respond to these changes.

Watershed Management: Terrain identification using neural networks can help in managing watersheds and water resources. By identifying the terrain features that impact water flows, such as slopes, vegetation cover, and soil type, water resource managers can design strategies that will help conserve and manage these resource.

3.Social:

Urban Planning: Terrain identification using neural networks can help in identifying suitable areas for urban development. By identifying terrain features such as slopes, soil type, and water bodies, urban planners can design development strategies that are environmentally sustainable and socially responsible.

Infrastructure Development: Terrain identification using neural networks can help in identifying suitable locations for building infrastructure such as roads, bridges, and dams. By identifying the terrain features that impact the stability and durability of these structures, engineers can design infrastructure that is safe and reliable.

Humanitarian Aid: Terrain identification using neural networks can help in planning and delivering humanitarian aid in disaster-stricken areas. By identifying the terrain features that impact accessibility, such as slopes and water bodies, aid organizations can design strategies that can overcome these barriers and reach affected populations.

Tourism: Terrain identification using neural networks can help in promoting tourism in regions with unique terrain features such as mountains, lakes, and rivers. By identifying the terrain features that make these areas attractive to tourists, tourism promoters can design strategies that showcase the natural beauty of these areas while ensuring their sustainable use.

4.Political:

Resource Allocation: Terrain identification using neural networks can help in identifying areas that are rich in natural resources such as minerals, forests, and water. This information can be used by policymakers to allocate resources and design policies that promote economic development and social welfare.

Border Security: Terrain identification using neural networks can help in identifying areas that are vulnerable to illegal border crossings or smuggling. By identifying terrain features such as hills, valleys, and water bodies, border security forces can design strategies that effectively monitor and control these areas.

Emergency Response: Terrain identification using neural networks can help in predicting and responding to emergencies such as natural disasters, accidents, or terrorist attacks. By identifying the terrain features that impact emergency response such as accessibility and proximity to medical facilities, policymakers can design strategies that ensure timely and effective response to emergencies.

5.Ethical:

Data Privacy: Terrain identification using neural networks relies on the collection and analysis of large amounts of data. This data can include sensitive information about individuals such as their location, movement patterns, and personal preferences. It is important to ensure that this data is collected and used ethically and that individuals' privacy is protected.

Bias and Discrimination: Neural networks can be prone to bias and discrimination if they are trained on data that is skewed or incomplete. It is important to ensure that the data used to train terrain identification neural networks is representative of the entire population and that the algorithms used do not unfairly discriminate against any group.

Transparency and Accountability: The use of terrain identification neural networks must be transparent and accountable. This means that the algorithms and data used must be open to public scrutiny, and the decisions made based on this technology must be explainable and justifiable.

Environmental Impact: Terrain identification using neural networks can have a significant impact on the environment, especially if it is used to promote unsustainable development or infrastructure. It is important to ensure that the use of this technology is aligned with environmental sustainability goals and that the impact of its use is carefully monitored and mitigated.

Social Impact: Terrain identification using neural networks can have significant social impacts, especially if it is used to allocate resources or make decisions that affect people's lives. It is important to ensure that the use of this technology is aligned with social justice goals and that the impact of its use is carefully monitored and evaluated.

6. **Health and Safety:**

Disaster Response: Terrain identification using neural networks can help identify areas that are vulnerable to natural disasters such as earthquakes, floods, and landslides. This information can be used to develop emergency response plans that ensure the safety of people living in these areas.

Hazard Identification: Terrain identification using neural networks can help identify hazardous areas such as steep slopes, unstable terrain, or areas prone to erosion. This information can be used to develop hazard maps and warning systems that can help prevent accidents and injuries.

Infrastructure Planning: Terrain identification using neural networks can help identify areas that are suitable for the construction of infrastructure such as hospitals, roads, and airports. This information can be used to plan infrastructure development in a way that ensures the safety and health of the people using it.

Environmental Health: Terrain identification using neural networks can help identify areas that are affected by environmental factors that can impact human health such as air pollution, water pollution, and exposure to hazardous materials. This information can be used to develop policies and interventions that protect public health.

System Requirements

H/W Requirements (details about Application Specific Hardware)

Inertial Measurement Units (IMUs): These are sensors that measure acceleration, angular velocity, and magnetic field strength in three axes. They would be attached to the person's lower limb to gather data about their movements.

Microcontroller/Microprocessor: A microcontroller or microprocessor would be required to receive the IMU data and process it to extract features for input into the deep learning models. It would also be responsible for running the models and making predictions in real-time.

Wireless Transceiver: A wireless transceiver would be needed to transmit the IMU data from the microcontroller/microprocessor to a remote server or computer for processing.

S/W Requirements(details about Application Specific Software)

- Python: This is a programming language that can be used for developing machine learning algorithms and handling data.
- Deep Learning Libraries: Machine learning libraries such as TensorFlow, Keras, PyTorch used to build and train the deep learning models.
- Data Preprocessing Libraries: Libraries such as NumPy, Pandas, or Scikit-learn may be used for data preprocessing, feature extraction, and transformation.
- Time Series Analysis Libraries: Libraries such as statsmodels, PyWavelets, or tslearn may be used for time series analysis.
- Visualization Tools: Data visualization tools such as Matplotlib, Seaborn, or Plotly may be used for data visualization, model performance visualization, and error analysis.
- Development Environment: An Integrated Development Environment (IDE) such as Jupyter Notebook
- Version Control System: Version control tools such as Git and GitHub may be used for version control and code collaboration.

- Operating System: The software should be compatible with the operating system used on the microcontroller or microprocessor, which may be Linux-based or another operating system suitable for IoT devices.

Results and Discussion

Concluding with the scores for each label for each test file in terms of precision, recall, accuracy, and F1. The final model's average accuracy, precision, recall, and f1 score are 89.62%, 79.1%, 81.8%, and 79.9%, respectively. Among the models that were previously developed, the final model has the greatest f1 score. Both figures show that the individual walks on solid ground, descends steps, then resumes walking on solid ground before ending up on grass. We are aware of the class imbalance in our dataset. So, it is not sufficient to just test accuracy, as our model can accurately predict the majority class without any further input. In order to produce the precision, recall, and F1 score values for each class, as well as the macro and weighted average, see table 4 below. The loss and accuracy graphs for the LSTM model's training and validation data are displayed

We load the test datasets that were used to evaluate the model using the confusion matrix and classification report.

APPENDIX A

References

[1] Sergiy Fefilatyev, “Algorithms for Visual Maritime Surveillance with Rapidly Moving Camera”, Doctoral dissertation, University of South Florida, 2012.

[2] Domenico Bloisi, Luca Iocchi, Michele Fiorini, Giovanni Graziano, “Automatic Maritime Surveillance with Visual Target Detection”, International Defense and Homeland Security Simulation Workshop (DHSS), September 2011.

[3] Domenico Bloisi, Luca Iocchi, Michele Fiorini, Giovanni Graziano, “Camera Based Target Recognition for Maritime Awareness”, Information Fusion (FUSION), 15th International Conference, July 2012.

[4]. Rodrigo Da Silva Moreira, Nelson Francisco Favilla Ebecken, Alexandre Soares Alves, Frédéric Livernet4, Aline Campillo-Navetti, “A Survey on Video Detection

and Tracking of Maritime Vessels”, IJRRAS 20, July 2014.

[5] Digi-key Corporation. http://www.digikey.com/

[6] Alldatasheet. http://www.alldatasheet.com/